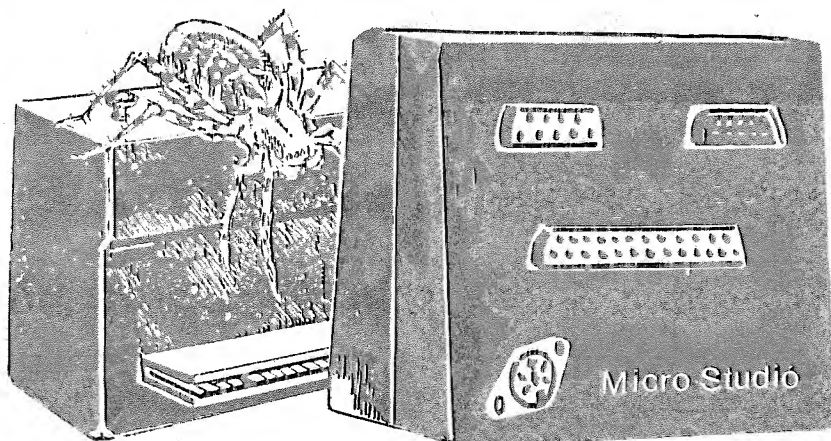


Tartalomjegyzék

1.	Verseny a javából	1
2.	Játék, POKE, térkép	2
2.1	Codename Mat (Micromega)	5
2.2	Universal Hero (Mastertronic)	8
3.	BASIC (A képernyő nagyítása)	14
4.	ENTERFACE (ENTERPRISE melléklet)	15
5.	SCREEN MACHINE ONE (Print'n'Plotter)	19
6.	Programozástechnika (A Beta Basic átalakítása)	22
7.	MULTIPRINT (Romantic Robot)	25
8.	128K (A hanggenerátor programozása)	27
9.	Gépi kód tanfolyam	29
+	Rejtvény	31



Felkerestük a legnagyobb hazai SPECTRUM Interface-ek gyártóját, a MICRO STÚDIÓ-t, hogy milyen érdekes, intelligens illesztő egységeket tudnak Olvasóinknak ajánlani. Az elhangzottakat az itt látható táblázatban ismertetjük.

Magyarázat:

* - fejlesztés alatt

programfék - olyan hardware egység, amely bármely program futását fokozat nélkül lassítja

C= - COMMODORE

Centr. - Centronics (párhuzamos)

meleg-reset - ilyenkor a 'lefagyott'

program legtöbbször visszaállítható

software-bank - külső EPROM-ból

- többnyire menükiválasztás alapján - az aktuális program gombnyomásra a RAM-ba tölthető

freezer - (fagyasztó) a program kimentése a pillanatnyi állapotában.

Visszatöltve a futás innen folytatódik.

turbo - sebesség: magnetofon esetében 3000 baud, C= floppy esetében 9600-19200 baud

színhelyes copy - olyan copy, amelynél a képernyő színei a nyomtatón a szürke különböző árnyalatában látszódnak.

Akit az itt ismertetett illesztők

részletesebben érdekelnek, azoknak

megadjuk a címet: MICRO STÚDIÓ

- 1536 Bp. Pf.: 323. Tel.: 820-832.

Köszönjük a tájékoztatást!

	MICRO-POKEer	PEAK-BOX*	ZX Interface 'C'
programfék	opcionális	beépített	beépíthető
periféria-kapcsolat	opcionális elcsatlakozó	elcsatlakozó	beépített RS232 Kempston
cél-output	magnetofon	magnetofon	C= floppy, C= printerek Centr. printerek
meleg-reset	van	van	van
software EPROM méret	beépített 8-16K	cserélhető modul 8-32K	beépített 32K
software-bank	opció 8-16K	bárm. program pl. copy, toolkit, stb.	tape-copy opció*
monitor	csak PEEK/POKE	tetszőleges dis/assemble kombinált	opcionális*
freezer (TOTAL RAM SAVE)	magnetofon normál/turbó	magnetofon normál/turbó	C= floppy normál/turbó*
hard-copy (képernyő másolat)	magnetofon	magnetofon	C= floppy C= printerek Centr. printerek RS232 M1:1, M2:1, színhelyes
Ár	2000,- Ft-tól	4000,- Ft-tól	11000,- Ft-tól

Szedés: LSI ATSz Info Osztály
A Spectrum Világ eddig megjelent részei utánvétellel
megrendelhetők a következő címen:
Spectrum Világ, Budapest-3, postán maradó, 1300

Felelős kiadó: Rucz Lajos, Székely László
K. F. eng.-sz.:

11 000 - Pátria Nyomda
Felelős vezető: Vass Sándor vezérigazgató

1. VERSENY A JAVÁBÓL

1

1988 nyarán is megrendezték Nagy-Britanniában a számítógépes játékok nemzeti bajnokságát (National Computer Games Championship). Ez a fogalom nálunk még ismeretlen, pedig a lehető legjobb terep arra, hogy összerázza a különböző számítógépek iránt érdeklődő fiatalok népes táborát.

Hazánkban még mindig elég nagy a szakadék a Commodore rajongók és a Spectrumosok között, s a legérdekesebb, hogy az egykor nagyhangú Commodore-os ill. a Commodore-osokra fújó Spectrumos ma már kéz- a kézben egymás mellett ül az IBM PC számítógépek előtt. Milyen érdekes, úgy hírlik, az IBM PC gépeken belül is igen széles a választék, az XT-n dolgozók vajon miért nem cikizik az AT-seket és fordítva? Ez a háború az utóbbi években csillapodott egy kicsit, de a magja még mindig a levegőben van. Talán bennünk van a hiba?

A brit verseny semmiféle viszályt nem tükröz, békésen elférnek egymás mellett a Spectrum és Commodore számítógépek tulajdonosai, mi több még választani is lehet, hogy az adott játékot melyik gépen szeretnénk teljesíteni.

Az idei verseny szponzora a US Gold cég volt, a lebonyolítást támogatta továbbá a számítógépes klubok brit egyesülete, a Personal Computer Show kiállítás rendezőse és a Newsfield Ltd. is, amely köztudott, hogy több ismert számítástechnikai sajtótermék kiadó-vállalata. A versenyen Sinclair Spectrum 128K +3 és Commodore 128D számítógépeket használtak fel, melyeket, a hozzájuk tartozó monitorokkal együtt ismert boltlánczatok (pl. DIXONS) és vállalatok biztosítottak.

A verseny a sportversenyekhez hasonlóan előfutamokból, városi, megyei döntőkből állt, az itt szereplő legjobb 15 joystick-gyilkos jutott tovább a három részes döntőbe. Az első részt Edinburgh-ban bonyolították le, a Fet Lore Boys Club asztalai mellett. A kiválasztott játék a US Gold - 720° c. játéka lett. Minden játékos 10 percet gyakorolhatott az éles küzdelem előtt. A verseny pontszámra ment, akik 100000 pontnál többet értek, már jutalmat kaptak (pl. software). Ennek a harmadnak a győztese 138660 pontot ért el.

A második harmad Nailsworth-ban zajlott. Hat-hét kiválasztott játékban elért minél több együttes pont megszerzése volt a cél.

A döntő csúcspontját - amelyet Birmingham-ban rendeztek - nagy várakozás előzte meg, ugyanis csak a verseny helyszínén derült ki,



hogy melyik játék vesz részt a versenyen. A Birmingham-i Highgate Sport & Leisure Centrum aulájában feszült csöndben hangzott el a 'Bionic Commando' c. játék neve, melyet nagy taps követett. A játékban a legjobb eredményt Commodore 128D számítógépen és KONIX típusú joystick-kel érték el (35990 pont).

Információink szerint hasonló versenyeket bonyolítottak le az utóbbi időben több nyugat-Európai országban is. Az ilyen szabadidős program megítélésünk szerint messzemenőig hasznos kikapcsolódás fiatal és idősebb számára egyaránt. Azt tanácsoljuk mindenkinek, aki személyi számítógéppel rendelkezik, vagy legalábbis elérhető számára, hogy otthon, munkahelyén, az iskolában kezdeményezzen hasonló programot, esetleg gyakorlásképpen próbálják ki otthon, hogy az említett játékokban elért bravúros pontszámot sikerül-e túlszárnyalni?

Zárszóként annyit szeretnénk megemlíteni, hogy a 'nevető harmadik'-ként emlegetett IBM PC számítógépekre is egyre több játék jelenik meg. Ez a géptípus kiváló semleges terepet biztosít olyan versenyek lebonyolítására, melyben a szereplők még mindig éreznek némi ellenszenvet, a másik gépével kapcsolatban. Egy IBM PC 'tetrís' vagy 'xonix' átmenetileg segít eloszlatni a hamis érzelmeket. Javaslatunkat azért mertük közzétenni, mert egyre több IBM PC kompatibilis gép kerül be az országba, s ezek többsége - sokszor a hozzáértés hiánya miatt - hosszú ideig kihasználatlanul várja billentyűi simogatását. Bár tudjuk, az IBM nem játékgép, ezt a tényt azonban megcáfolja az a több tucatnyi játék, amelyet a mai napig fejlesztettek rá, s a jövő játéakai is hasonló, professzionális számítógépekre fognak készülni.

Brainstorm - Firebird

Ez egy stratégiai játék, melyben a játékosnak az elméjét és reflexeit kell próbára tennie. Cél a lehető legrövidebb idő alatt minél több pontot szerezni. Ez úgy érhető el, hogy a mozgatható színes vonalak segítségével egy golyót kell a képernyőn körben elhelyezkedő piros területekben csapdába ejteni. Kezdetben beállíthatjuk a kezdő-képernyő számát és a nehézségi szintet is. 26 A-Z-ig jelölt képernyő van, ezek egyre nehezebbek. A játéktér színes négyzetekre van felosztva, a játékos 3 színes vonalat és egy kurzort mozgathat. A négyzetek pirosra, zöldre, bíborra, kékre vagy feketeire vannak színezve. Piros négyzetet megerintve nő a pontszámunk, a bíbor négyzetek csökkentik pontszámunkat, a zöldek eltérítik a golyót a képernyő más területére, a fekete és kék négyzetek semlegesek. Egy vagy több vonal segítségével (melyek a kurzort a kívánt négyzetre felé mozgatva, és tüzelve használhatók), a golyót csapdába kell ejteni. Cél a gép által kitűzött pontszám elérése. Ha a megfelelő pontszám összegyűlik, a játék a következő pályán folytatódik, és előző pontszámunk átiródik. Pontszámunk csökkenhet is, s ha eléri a nullát, vagy ha a pályát nem teljesítettük, a játéknak vége.

Frightmare - Cascade

Minden ember tudatalattija tartalmaz ősi elemeket, melyek belopóznak álmainkba. A 'Frightmare' lehetővé teszi a játékosnak, hogy ebbe a pszichikai mélységbe lépjen be és átadja magát a rettegésnek, a legsötétebb képzelgések földjén. A lieder-cnyomás 8 és fél óráig tart, a 4 álmózná szuper-természetes környezetében, végighaladva egy romos, málo emlékművekből, facsonkokból és vérvörös koponyákkal teli szobákból álló területen. Mint az igazi álomban, itt is hatalmasakat lehet ugrani - leesni, ill. meghalni, lehetetlen. Ezen az 'álmok földjén' őt ősi mese és legenda áldozatai kísértének. Ha a játékos valamelyikükkel érintkezésbe kerül, őt élete közül elveszít egyet. Zombik és skarlátvörös, test nélküli kezek jelennek meg, melyek az összegyűjthető fegyverekkel alháríthatók. A szenteltvíz a szörnyek mindegyikét azonnal megöli, a revolvereknek (melyekhez töltényt is kell gyűjteni) és a feszületeknek a hatása szelektív. Más egyéb eszközök növelik az ugrási képességet, plusz életet jelentenek és lehetővé teszik, hogy különböző szobákba jussunk be. A játék rendkívül érdekes logikája vonzó, de a gyengébb idegzetűek inkább válasszanak helyette mást.

Magnetron - Firebird

A Magnetron egy aranyos robot-ötlet, hagyományos Firebird megoldásokkal. A KLP-2 egy kis droid, amely már korábbi játékokban is szerepelt. Ez alkalommal ördögi droidok keringenek a Quarteck bolygó körül, melyet nyolc felfegyverzett műhold fenyeget. A KLP-2 az első műhold fedélzetére kerül, ahol feladata, hogy a sokszínű és három-dimenziós rámpákból és utakból álló területen megtalálja a műhold 4 reaktorát, majd hatástalanítsa ezeket. A reaktor stabilitása a pozitív üzemi-angrudaknak és a negatív fékezorudaknak az egyensúlyától függ. A KLP-2 az üzemi-angrudak cserélgetésével a töltést megváltoztathatja, s ez a reaktor beállítását vagy túlterhelődését eredményezheti. Az állomás droid lakói rossz szemmel nézik küldetését, s bár a KLP-2-nek nincs elég tüzeje ahhoz, hogy közvetlenül elbánjon velük, viszont hozzá tud kapcsolódni más droidokhoz, hogy tulajdonságaikat befolyásolja. Egy sikeres összekapcsolódás hatására leveti páncélzatát, mely végül is egy új életet jelent. Ha mind a 4 reaktort sikerült leállítani, a műhold kikapcsol, és lehetővé teszi, hogy a KLP-2 átkerüljön a következőre. Izgalmas akció, szép grafikai kidolgozással.

Sidearms - GO!

A GO! legfrissebb játékát az un. 'agresszív' játékok kedvelőinek ajánljuk. Bozon, a zsarnok és megalomániás, a Föld kiirtásával fenyeget. Azonnali megtorló akcióra van szükség, és Henry hadnagyot, ill. Sanders őrmestert jelölik ki a feladatra. Céljuk az, hogy beszivárognak Bozon földalatti birodalmába, és megsemmisítsék az ellenség végső fegyverét. A játékban 1 vagy 2 játékos menet is választható. A játékos különböző, vízszintesen mozgó fekete-fehér háttérrel folytatja küzdelmét. Akciónkról az ellenfél is értesült, s azonnal riasztotta védelmét. A harcosok különböző formákban jelennek meg. Némelyik megsemmisített ellenfél jutalompontot hagy maga után, ezeket össze lehet gyűjteni. A tokok alapvetően a tüzerőt és a fegyverzetet növelik, de mindegyik további segédeszközt is jelent a küzdelemben. Meghatározott számú lövés a tokot egy másik fegyverre alakítja át. Egy lövéssel nő a sebességünk, 4 lövéssel páncélóköll lesz, stb. Fegyvertárunkat ilyenkor a billentyűzetről váltogathatjuk. Minden szint végén egy lezárt okádó szörnyeteket kell legyőznünk. A játék végén pedig Bozon végső fegyverét kell megsemmisítenünk. Hagyományos grafika, sajnos hiányzó ötletekkel.

Antiriad

A sérthetetlenség bevitelét a 142/3742/40000 file-térképű programon mutatjuk be.

Töltsük be a loader-t, majd RESET. Írjuk be a következő programot:

10 CLEAR 24999: LOAD"" CODE: RANDOMIZE USR 28350: LOAD"" CODE: POKE 54535,0: POKE 54536,0 POKE 54537,0: RANDOMIZE USR 45600

Futtassuk a programot (RUN), és indítsuk tovább a magnót.

Leonard

Az örökelet bevitelét a 241/6916/20000/20536/1704 file-térképpel rendelkező verzióra közöljük.

Az örökelethez a 64309-es címre 0-at kell írni.

Töltsük be MERGE-dzsel a loader-t, és írjuk be a következő sorokat:

20 CLEAR 24999: FOR i=23834 TO 23844: READ a: POKE i,a: NEXT i: RANDOMIZE USR 23760
30 DATA 175,50,53,251,61,55,229,195,86,5

Futtassuk a programot, és indítsuk el a magnót!

Sabre Wulf térkép

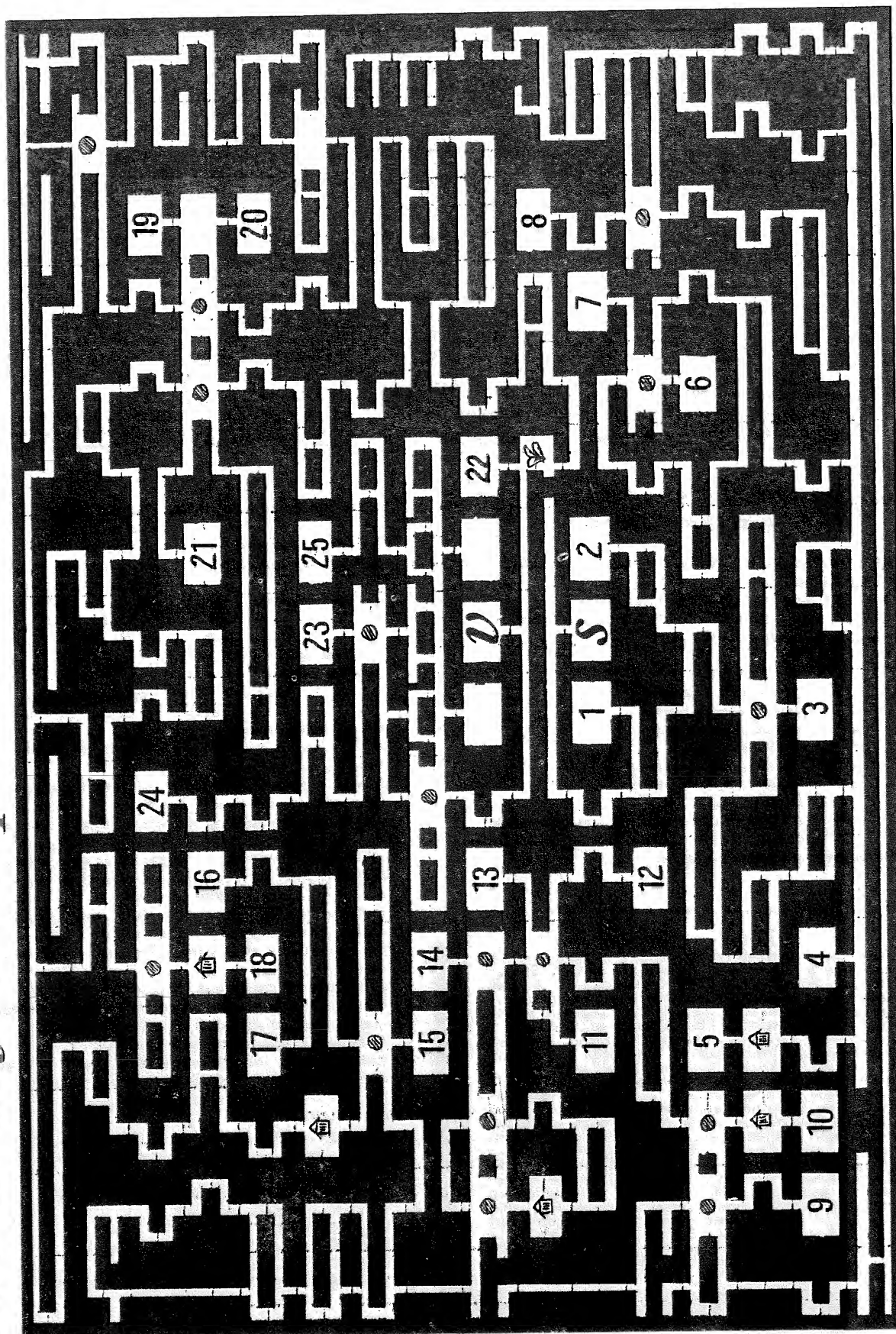
S start
V vége

virágok

palack

1-25 pajzsok

kunyhó



Sophistry - CRL

A Sophistry egy 21 szintből álló komplexumban játszódik. Minden szint néhány 3 dimenziós, egymással összeköttetésben álló blokkot is tartalmaz. A játék kezdetén a szerkezetnek csak a 20 %-a nyitott. A játékos célja, hogy megszerezze a körben elhelyezkedő 64 kulcsot, melyek nyitják a 21. szintre vezető kaput, valamint biztosan kinyitják a szerkezetet. Ezek a kulcsok megszerelhetők közvetlenül, vagy pontokat gyűjthetünk, és kereskedés útján kell megszerezni őket. Kis tok alakú járművel haladunk, ez 4 irányban tud mozogni. A közvetlenül pontot érő és célul kitűzött blokkok növelik pontszámunkat. A blokkokon és jokereken lefelé és felfelé különböző sorrendben áthaladva is növeljük pontszámunkat. Ha elvesztünk egy blokkot, vagy jeletlen kijáraton próbálunk kimenni, akkor elvesz egy életünk. A keresés alapelve összefügg a blokkok sajátos tulajdonságával. Baloldalon jelenik meg, hogy bizonyos korlátozások érvényesek-e, vagy sem. Némely blokknál a kijáratok zárva vannak, és bizonyos pontszámig ki sem nyílnak. Véletlenszerű furcsa dolgoktól sem kell meglepődnünk. A képernyőn látjuk, hogy éppen milyen szinten vagyunk, mennyi életünk van, az idő és a pontszám is. Érdekes logikájú ötlet.

Thing - Players

A Players játéka a Code Masters-hez hasonlóan meglepően alacsony áron kínálja a szigetországban, ez nem azt jelenti, hogy a színvonal is alacsony. Ismét egy sci-fi történet feldolgozását végezték el a szerzők. A Dingalingerek bolygóját nukleáris háború fenyegeti. Ball-t - aki az Urközi Harmónia Bizottság képviselője - és segédjét THING-et - aki intergalaktikus őrszem - elküldték, hogy tárgyaljanak a Dingalingerek vezérével. Oda-érkezésükkor a vezér megtagadta a Ball-lal való tárgyalást, ezt attól tette függővé, hogy THING elvégez-e 10 fontos feladatot. THING, Ball társaságában mindössze egy telefonkönyvvel felszerelve a bolygó 26 szintjén halad keresztül, melyeket telefonfülkék sora köt össze. Lebegő kapszulák sora tartalmazza a dolgokat, melyeket össze kell gyűjteni, s ezeket megfelelő kombinációban használni kell. Munkájukat a Dingalingerek megpróbálják akadályozni, viszont Ball rendelkezik kábitőfegyverrel. Ha az 5 nap alatt minden rejtélyt megfejtenek, akkor a vezér rossz hangulata eloszlik és minden megoldódott, máskülönben a kiút a halál. Bonyolult, gondolkodtató akciójáték, szép megoldásokkal.

GARY LINEKER'S SUPERSTAR SOCCER (Gremlin Graphics)

Kurzor irányítása: k - lefelé, o - felfelé, 0 - aktivizálás
League Status: jelzi

- pénzünk állását - játékidőt (szezont)
- szezon időt - lejátszott bajnoki meccsek számát
- lejátszott kupameccsek számát

Csapat történet:

- YR - aktuális év(ek)
- W - győzelmeink száma
- D - döntetleneink száma
- L - vereségeink száma
- PT - pontszámunk
- GF - rugott góljaink száma
- GA - kapott góljaink száma
- DV - aktuális liga
- DP - a mi ligánkban hányadikként végeztünk
- RT - összesen a IV ligában hányadikak voltunk

Mérkőzések eredményei: fel/le

Zolyx - Firebird

A Firebird játéka a többségében nem szokott csalódní a színvonalas játékokat kedvelő Spectrumos. Így van ez a ZOLYX esetében is. A játék ötlete - a Tetrishez hasonlóan - egyedi, némi logikai gondolkodást is igényel (IBM gépeken XONIX néven vált ismertté). Célunk, hogy az üres képernyő részeit egyenes vonalakkal bezárjuk. A bezárt területek beszíneződnek. A vonalakat a kurzorral húzhatjuk, s eközben ügyelnünk kell arra, hogy a véletlenszerűen pattogó golyókat elkerüljük. Ezek a golyók nem érhetnek a kurzorhoz, s egy befejezetlen vonalhoz sem, mert egy életünk oda lesz. A folyamatos vonal hosszúságát meghatározza egy vérmes ellenség is, mely egy idő után üldözni kezdi a kurzort. Ha ezzel a mimózával összelelkeztünk, ismét elvesz egy életünk. A játék során minden szinten a képernyő legalább 75 %-át be kell festenünk. Ebben a műveletünkben meghatározó szerepe van a vésszenes lekettyegő időnek is. 75 % felett minden egyes százalékkért jutalompontot kapunk. Szintenként eggyel nő a képernyőn bolygogó golyók száma is. Kiváló ügyességi játék, elsősorban házi versenyek lebonyolításához ajánljuk.

Ligák állásának áttekintése: az ellenfelünket és minket jelez a sáv

Betöltés/kimentés: kazettás magnetofonról/ra

Játékosok összeszedése: név/poszt/erősség/életkor for-mában

A poszt jelzései: CF - kapus LW - bal csatár
LD - bal hátvéd RW - jobb csatár
RD - jobb hátvéd GK - irányító

Az erősség max. 50 lehet.

Játékos csere:

Return - vissza Try Trade - csere

New Teams - Új csapat lehívása

Csapatok lehívása:

Aktivizálás után a fel/le billentyűkkel megtekinthetjük a csapatok játékos-állományát

Ei kell végezni az általunk adott, majd a kért játékos kiválasztását, és a ráfizetés esetleges összegét is. Az ezt követően villogó szöveg azt jelenti, hogy folyik a tárgyalás. Kis idővel később megjelenik egy szöveg, amennyiben ez fehér, akkor sikerült, ha vörös, akkor nem sikerült a csere. Felállítás: Jelzi a teljes játékos állományt és a kezdőcsapatot. A cserék zölddel jelzettek. Ki kell választani, hogy melyiket akarjuk lecserélni. Move segítségével játékosok között, swap segítségével a csapaton belül cserélhetünk. Ha egy játékos neve vörössel jelzett, azt feltétlenül le kell cserélni.

Szerződés a csapathoz: Szerződést bármikor megszakíthatunk. Meg kell adni az ország nevét, a szezon számát, majd ki kell választani a megfelelő csapatot.

A mérkőzés elindítása (NEXT GAME):

Game Speed - hányszorosra legyen a játék felgyorsítva
Centre Coach - irányítás (a számítógép és a játékos összes variációjában)

OK - indítás

A számítógép-számítógép verzió esetén azt is meg kell adnunk, hogy végig akarjuk-e nézni a meccset.

Üzenetek:

YOU GO THROUGH TO THE NEXT ROUND -

továbbjutottunk a kupában

NEW SEASON - Új szezon

PROMOTED - Az új ligába felkerültünk

DEMOTED - kiestünk a ligából

... WON THE CUP - megnyerte a kupát

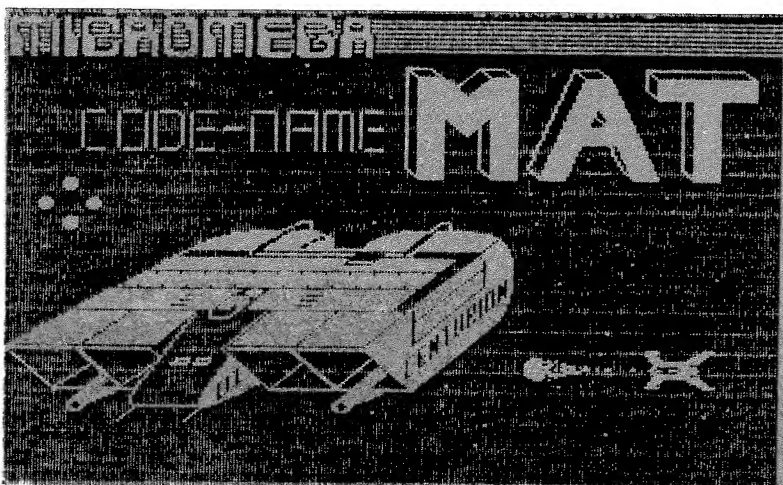
2.1 CODENAME MAT

5

Folytatjuk jól megkezdett hagyományunkat, miszerint 'elfelejtettnek' hitt régebbi játékokat is előveszünk a fiókból, elsősorban tömeges igények kielégítése céljából.

A CODENAME MAT érdekes kivitelű MICRO-MEGA játék, amely egy a fantasztikus űrben játszódó történetek hosszú sorából. Talán azért választottuk most ezt a régebbiek közül, mert kezelése bonyolult, és a maga idejében nagy sikert aratott.

Történt egyszer, hogy a MYON-ok uralmuk alá akarták venni a naprendszert. A naprendszer védő-flottái a szétesés határán álltak, erők teljesen kimerültek. A MYON-ok tudomására jutott, hogy a földön kifejlesztés alatt áll egy új űrhajó, melynek prototípusa az USS CENTURION már elkészült. Ez a hadihajó képes egyedül a MYON-ok feltartóztatására, ezért a MYON-ok bevetettek mindent, hogy az USS CENTURION ne indulhasson támadásba. Az USS CENTURION és egyben a teljes Bolygóközi flotta irányítása túlszeli az emberi értelem határán, ezért kifejlesztettek minket a robotot, a vezérlési feladatok ellátására. A MAT nevet kaptuk, s feladatunk, hogy az USS CENTURION-t irányítva szabadítsuk meg a naprendszert a MYON-októl.



A játék betöltését követően megjelenik a főmenü, mely három lényegesebb opciót tartalmaz. Kiválaszthatjuk az irányítást (billentyűzet/joystick). Ami azt illeti kivételesen azt tanácsolhatjuk válasszuk inkább a billentyűzetet, ugyanis a vezérlés nem könnyű dolog. A billentyűk kiosztása a tevékenység-csoportok bontásában a következő:

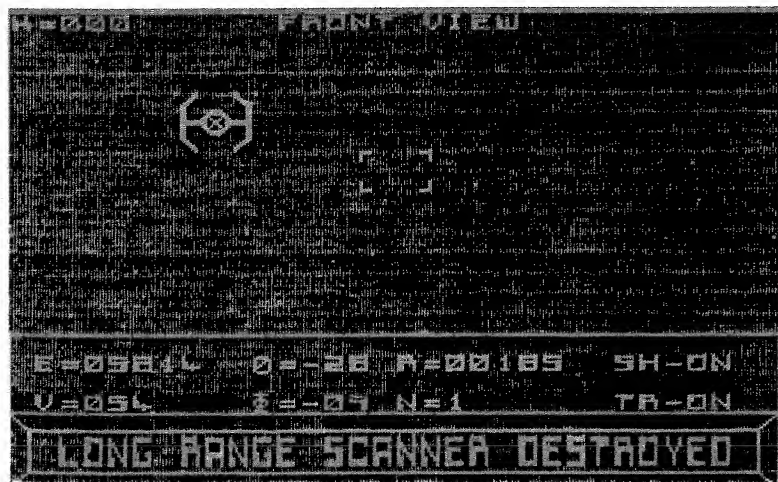
Irányítás és tüzelés	Hajtómű-irányítás	Display	Egyéb
6 - balra	1 - lassítás	F - képernyő előre	W - vonósugár
7 - jobbra	2 - gyorsítás	R - képernyő hátra	D - védőpajzsok be/ki
8 - le	3 - fékezés megállásig	L - hosszú távú radar	K - követő computer be/ki
9 - fel	4 - utazósebesség	S - szektor radar	A - követett ellenség beállítása
Q - tűz	5 - max. sebesség	C - naprendszer térképe	T - parancsok közvetítése
		H - program megállítása/újraindítása	

A vezérlés beállítása mellett kiválaszthatjuk a játék nehézségi szintjét:

1. Rövid gyakorló játék
2. Teljes játék, közepes méretű MYON inváziós flotta ellen
3. Teljes játék, nagyméretű, gyorsan mozgó MYON inváziós flotta ellen

A menü harmadik pontja segítségével a vezető típusát választhatjuk ki:

1. Parancsnok (mi irányítjuk a Bolygóközi védő-flotta egységeinek telepítését)
2. Pilóta (a flották telepítését a számítógép végzi)



Amikor elkezdjük a játékot, a pilótafülke jelenik meg, valamint kilátás előre. Mozdulatlanul állunk a sejtelmes és bizonytalan térben. Először is nem árt, ha 'D' segítségével aktivizáljuk a védőpajzsokat (soha nem lehet tudni), gyorsuljunk fel utazósebességre (30%) a '4' billentyű megnyomásával, hiszen órákig várhatunk, mire történik valami. Közben néha lessünk ki hátrafelé az 'R' segítségével, a MYON-ok kiszámíthatatlanok. Tanulmányozzuk a műszereinket, 'K' nyomtatása esetén aktivizálhatjuk a nyomkövető számítógépet, ez automatikusan első, vagy hátsó display módba vált, ha valahol ellenséget észlel. A szektor-radart is tekintsük meg, ez 'S' megnyomásával lehetséges. A szektor-radar megmutatja a helyzetünket a bolygórendszerben, jelzi a flották, hordak helyzetét is. A naprendszer térképét a 'C' megnyomásával kérhetjük le. Ez kevésbé részletes módon megmutatja a bolygórendszert, a szövetséges

flották erejét és űrhajónknak a USS CENTURION-nak az állapotát. A képernyő alsó részén található az ún. csata-computer, ami állandóan változó képet mutat. Itt találjuk még a bolygóközi rádiót is, amely információt közvetít a bolygók támadási helyzetképéről, károkról, valamint figyelmeztet minket, ha ilyen van a közelünkben. Ha a közelünkben található. Ha a bolygórendszerben más területre akarunk eljutni, használnunk kell az ún. vonósugarat. Ez a 'W' billentyű megnyomásával aktivizálható. Másik bolygórendszerbe való átjutás csak a csillagkapun keresztül lehetséges. Ezt meg kell keresnünk, és keresztül kell repülni rajta. A kilépő csillagkaput vörös, a belépőt pedig világoskék szín jelzi.

Ha parancsnok módban játszunk, akkor utasításokat közvetíthetünk az irányításunk alatt álló flottáknak, a 'T' billentyű lenyomása segítségével.

Háromféle MYON támadó hajóval találkozhatunk, sajnos ezek minden körzetben különböző kombinációban és számban fordulhatnak elő:

- **cirkáló** (akkor támad, ha egy körzetben legalább kettő van belőle)
Sebessége: 20 % ; Védőpajzs: nincs
- **kísérő cirkáló** (akkor támad, ha 2000 egységgel közelebb van, egyébként egy meghatározott utat jár)
Sebessége: 60 % ; Védőpajzs: nincs
- **csillagromboló** (azonnal támad)
Sebessége: 80 %
Védőpajzs: regenerálódik.
Ha megsemmisül a pajzs, vörössé válik és a csillagromboló kb. 2 percig menekülni fog, ugyanis ennyi idő kell a pajzs regenerálódásához.

Egy bolygó vagy hold akkor van támadás alatt, ha a bolygó körzetével szomszédos 8 körzetből négyet már elfoglaltak a MYON-ok. Körülbelül három percünk van, hogy ezek közül egyet megtisztítsunk, mielőtt a MYON-ok lerombolnák céljukat. Egy bolygó elfoglalása esetén a MYON-ok nyerne két cirkálót és egy csillagrombolót, vagy két cirkálót és egy kísérő-cirkálót, amennyiben holdat romboltak le. Néha jobban megéri, ha a bolygót inkább leromboljuk, mint ha megvédenénk. A MYON-ok nem egészen logikusan gondolkodnak, így az idő nagy részét a bolygók támadásával töltik el, ahelyett, hogy a csillagkapukon átrepülve azonnal a Föld közelébe jussanak (kivételek, ha már nagyon vesztesre állnak). A USS CENTURION hamar megsérülhet egy-egy támadás alkalmával. Ezeket a sérüléseket ki tudjuk javítani, ha elérkezünk egy bolygóra, vagy egy holdra. Egy égitestet akkor érünk el, ha megállunk a közelében, attól kb. 400-500 egység távolságra, tőle kevesebb mint 5 fok függőleges ill. vízszintes szögben. A CENTURION megsérül, ill. meg is semmisülhet, ha túl közel megyünk az égitesthez. Mielőtt újra útnak indulunk, el ne felejtjük a védőpajzsokat ismét aktivizálni, de legfőképpen a megfordulásról ne feledkezzünk meg. Ha sérülés ér bennünket, ez különbözőképpen lehet hatással az űrhajó állapotára. A sérült hajtóművek egyenetlenül dolgoznak. A többi rendszer megszűnik működni, ha találat éri. Számos rendszernek az elvesztése komoly hatással van a hajó navigációs képességeire. A vonósugár sem szenvedhet károsodást, mert működésképtelen lesz. Ha a CENTURION energiája elfogy, vagy megsemmisül, ill. ha a Föld semmisül meg, akkor a háborút sajnos elvesztettük. Az utolsó MYON űrhajó megsemmisítése jelenti a győzelmet. Pontszámunkat többféle szempont szerint állítja össze a program: mennyi időt teljesítettünk, mennyi bolygót ill. holdat mentettünk meg, milyen játék-szinten játszottunk, ill. milyen hatékony volt a harci technikánk.

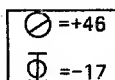
A következőkben részletesen áttekintjük a harc lebonyolításában jelentős szerepet játszó eszközeinket.

CSATA-COMPUTER (BATTLE COMPUTER)

Állandóan a képernyőn található.

energia → E=08721

sebesség(%) → V=030



← a követett objektum
szögeltérése az egyenestől
← a követett objektum
fel/le szögeltérése

R=15721 ← objektum távolsága
N=2 ← objektum száma

SH-ON ← védőpajzs be/ki
TR-OFF ← követő computer be/ki

NAGY HATÓTÁVOLSÁGÚ RADAR (LONG-RANGE SCAN) - 'L' billentyű

Bemutatja a teljes körzet három dimenziós képét (ahhoz képest, ahol tartózkodunk). Minden körzetet egy kör alakú (gömb) univerzumként kell elképzelni. Ennél fogva egy űrhajó, amely pl. hátul eltűnik, elől fog megjelenni. A nézeti kép felülnézet. A CENTURION-t egy rögzített középső pont jelzi, orrával a képernyő felső része felé. Ebből adódóan, ha balra fordulunk, akkor az egész képernyő (úgy tűnik) jobbra fog fordulni a pozíció körül. Ha hurokban repülünk, ('bukfenc'), akkor a képernyő függőlegesen fog forogni. Érdekes addig foglalkozni vele, amíg megértése már csak egy másodperc töredéke lesz csupán.

SZEKTOR RADAR (SECTOR SCAN) - 'S' billentyű

Ez valójában egy 10x7-es négyzetháló, melynek minden része egy körzetet jelöl. Az egyes jelölések a következők:

- - a rendszer fő-bolygója
- ⋄ - csillagkapu (piros:ki, világoskék:be)
- ⊞ - USS CENTURION
- - hold
- ≡ - 3 MYON-ból álló flotta
- + - vonósugár beállító kurzor
- F1 - Bolygóközi védő-flotta
- ≡ - 2 MYON-ból álló flotta
- - 1 MYON-ból álló flotta
- ... - a pontok az ép hajókat jelölik

NAPRENDSZER TÉRKÉP (SOLAR CHART) - 'C' billentyű

A naprendszer térkép mind a 7 bolygórendszer képének együttes miniatürizált és az óramutató járásával ellentétes irányban 90 fokkal elforgatott képéből áll.

Az itt található jelölések:

- - MION flotta
 - + - USS CENTURION
 - X - Bolygóközi védő-flotta
 - - bolygó
 - - hold
- Alatta a CENTURION állapotát jelző műszereket találjuk:

PH1 PH2 SHE LME BCO SRE STR LRS
fotonagyúk pajzsok hajtómű csata-comp. bk.rádió bk.adó nh.radar

Ha a műszer kódjele villog, akkor sérülést jelez, ha vörösre vált, akkor megsemmisült.

VONÓSUGÁR (WARP DRIVE) - 'W' billentyű

Ha egy másik körzetbe akarunk eljutni, kapcsoljuk be a SZEKTOR RADAR-t az 'S' billentyű segítségével. Az irányító billentyűkkel vezessük a vonósugár beállító kurzort arra a körzetre, ahová el szeretnénk jutni. Nyomjuk meg a 'W' billentyűt, és a vonósugár bekapcsolódik. A képernyő az előre történő kilátást fogja mutatni és a vonósugár csatornát jelző gyémántot. Amint keresztülrepültünk a vonósugár csatornán, kövessük a gyémántot, úgy, mint ha egy űrhajót

2.1 CODENAME MAT

7

Üldöznénk, mert máskülönb en letérünk az útról. A vonósugár beállító kurzort előzetesen is beállíthatjuk, és a vonósugár használatát későbbre ütemezhetjük.

CSILLAGKAPUK (STARGATES)

Ha egy másik bolygórendszerbe akarunk átjutni, keresztül kell repülnünk a csillagkapun. Egy világoskék színűn keresztül juthatunk be a rendszerbe, és egy vörös színűn keresztül ki. Először is el kell jutnunk a csillagkapu körzetébe, azután meg kell keresni a képernyő, a nyomkövető, vagy a csata-computer segítségével. Már kb. 2500 egység távolságból észlelhető. A bolygóközi ugráshoz a csillagkapu középpontjába kell repülni. Minél gyorsabban repülünk, annál pontosabbnak kell lenni. A következő bolygórendszerbe a csillagkapu másik végének körzetében fogunk megérkezni. Ha a csillagkapunál rajtunk kívül még egy MYON is jelen van, nem használható, egészen addig, amíg a MYON-t meg nem semmisítjük.

BOLYGÓKÖZI ADÓ (SUBSPACE TRANSMITTER) - 'T' billentyű

Ha parancsnok módot választunk, utasításokat kell közvetíteni a Bolygóközi védő-flotta számára, hogy a stratégiánknak megfelelően helyezkedjenek. Ehhez először is meg kell nyomnunk a 'T' billentyűt. Tudnunk kell a flotta nevét, ahova közvetíteni szeretnénk. Adjuk meg a bolygó betűjét, amelyhez ez tartozik, majd a flotta számát, és nyomjuk meg az ENTER-t. Pl. az 'S2' a Szaturnusz flotta, ez a kettes számú, vagy 'U1' az uránusz flotta, ez 1-es. Most gépeljük be az üzenetet és ismét nyomjunk ENTER-t. Az üzenetnek tartalmaznia kell az irány első betűjelét (angolul), amerre mozogni akarunk, a távolságot, szükség szerint többször is, és ezt még egy képzővel megtoldhatjuk. Az 'R' képző azt jelenti, hogy a flotta maradjon az új pozíciójában. A 'D' képző a körzet kerületének a védelmére szólít fel. Pl. egy 'E2N5R' üzenet azt jelenti: menj keletre 2-t, északra 5-t és maradj ott. 'W5D' jelentése: menj és véd meg a nyugatra elterülő 5. körzet kerületét. Minden flotta emlékezik az utolsó 'R' vagy 'D' parancsra és ha újabb üzenetben nem szerepel képző, úgy azt hajtja végre. A flotta nem tud keresztülhatolni egy bolygó vagy hold körzetén. Hasznos lehet 'R' utasítást adni és ezzel a flottát megállásra kényszeríteni, mialatt egy üzenetet állíthatunk össze. Ha egy flotta az útja során MYON-okkal találkozik, azonnal harcba bocsájtkozik és nem veszi figyelembe az újabb utasításainkat. Az 'A' képzőt használva megakadályozhatjuk azt, hogy a flotta utasításunk nélkül harcba keveredjen a MYON-okkal. A szektor radaron ellenőrizhetjük adott utasításainkat.

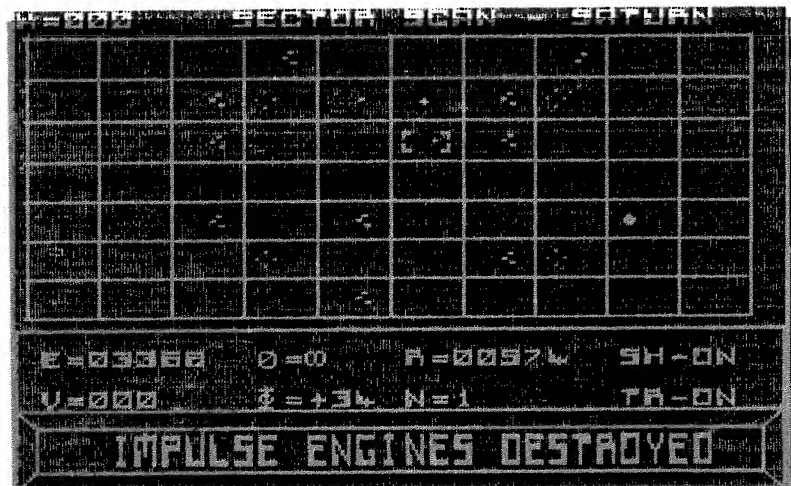
NYOMKÖVETŐ SZÁMÍTÓGÉP (TRACKING COMPUTER) 'K' billentyű

A 'K' billentyű segítségével lehet be illetve kikapcsolni. Automatikusán fogja váltogatni az első és hátsó képernyőket, igazodva a követett objektum helyzetére. Mindig a képernyő (első/hátsó) irányába tüzelhetünk. A követett objektumot a csata-computer 'N' karakter utáni kódja jelzi. A 'P' bolygót, az 'S' csillagkaput, a 'O' objektum hiányát, míg '1', '2' és '3' a MYON(ok) számát jelzi, amelyiket követni szeretnénk a körzetben. A támadók közül az 'A' billentyű segítségével választhatunk. A csata computer távolság-, ill. szög kijelzői a követett objektumra vonatkoznak. Amikor először belépünk egy MYON-ok által elfoglalt körzetbe, érdemes kiválasztani a MYON-ok közül a legközelebbit, mert másként könnyen meglepetés érhet bennünket.

ÁLTALÁNOS INFORMÁCIÓ

Ha harcba szállunk, először a cirkálókat semmisítsük meg, méghozzá azt amelyiket követjük, máskülönb en előfordulhat, hogy akkor kerül űrhajónk elé egy cirkáló, amikor a nyomkövető számítógép hátsó képernyőre kapcsol. A kísérő-cirkálók üldözése időigényes feladat. Néha jobb a kísérő-cirkálókat a végére hagyni, kivéve, ha közvetlen fenyegetést jelentenek egy bolygóra. Az űrben aszteroidákat is találunk. Követni persze nem tudjuk ezeket, de kárt okozhatnak - figyeljünk ezekre! A harcot nem bölcS folytatni, ha a csata computer, vagy a hosszútávú radar megsérült. Egyik sem használható hatékonyan bolygók vagy MYON-ok megtalálásához, de mindkettő elvesztése azt eredményezi, hogy csak a nyomkövető computer képernyő-váltogatása segít egy-egy objektum helyzetének meghatározásában.

Tartsuk figyelemmel az energiaszintet. A védőpajzsok, a vonósugár, a hajtóművek és a fotonagyúak különböző mértékben fogyasztják az energiát. A legsúlyosabb energiavesztéseket a MYON-ok fotonagyúinak találatai okozzák. 1000 egység alatt a képernyő villogni kezd.



Reméljük ismét kedveztünk az űrhajós-stratégiai-kaland játékok kedvelőinek. Bár a játék 4 éve készült, ettől függetlenül joggal sorolhatjuk a maga nemében is színvonalas játékok sorába.

Holiday in Sumaria

Az örökélet bevitele a 146/6916/20000/20536/1704 file-térképpel rendelkező programokra érvényes.

Az örökélethez az 50330-as címre kell zérust elhelyeznünk.

Töltsük be a loader-t MERGE"" segítségével, és írjuk be a következő BASIC sorokat:

20 REM (ENTER), majd POKE 23757,83 (ENTER)

20 CLEAR 24999: FOR i=23823 TO 23832: READ a: POKE i,a NEXT i: RANDOMIZE USR 23760

30 DATA 175,50,154,196,61,55,229,195,86,5

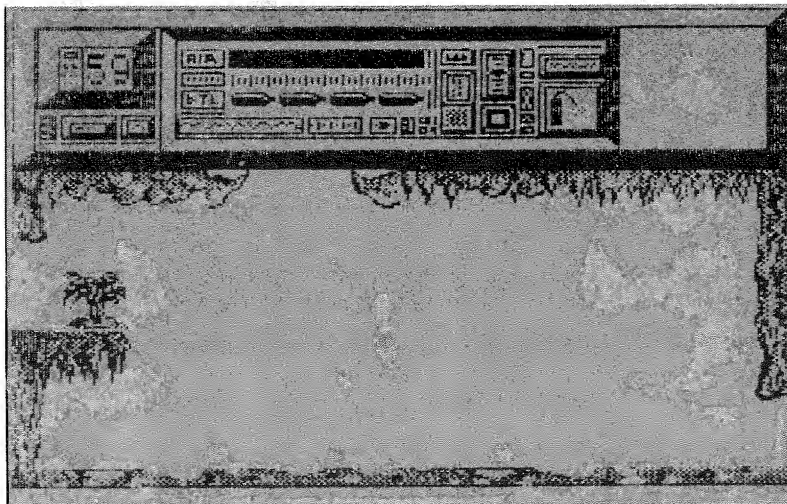
Futtassuk a programot és indítsuk el a magnetofont.

Az utóbbi időben több Mastertronic játék ismertetését vállaltuk, mert úgy érezzük, hogy ezeknek a játékoknak a színvonala megérdemli, bonyolultsága pedig megköveteli, hogy foglalkozzunk velük. Szívesebben vállalkozunk az ilyen stílusú játékok megoldásainak leközlésére, mint az egyértelmű akció-játékok ismertetésére (pl. URIDIUM).

Egy XXI.sz.-i űrkomp békésen száguldott tova az űrben, de véletlenül belesodródott egy előre kiszámíthatatlanul közeledő aszteroida-mezőbe, és meghibásodott. Sajnos az üzemanyag-tartály is ki-lyukadt. A vész helyzetben még annyi módosítást tudunk végrehajtani, hogy az űrkompot a fedélzeti számítógép segítségével bolygó-körüli pályára állítsuk. Ennek a feladatnak a végrehajtására mindössze 60 perc áll rendelkezésünkre, mert különben az űrkomp egyenesen nekiütközik a bolygónak.

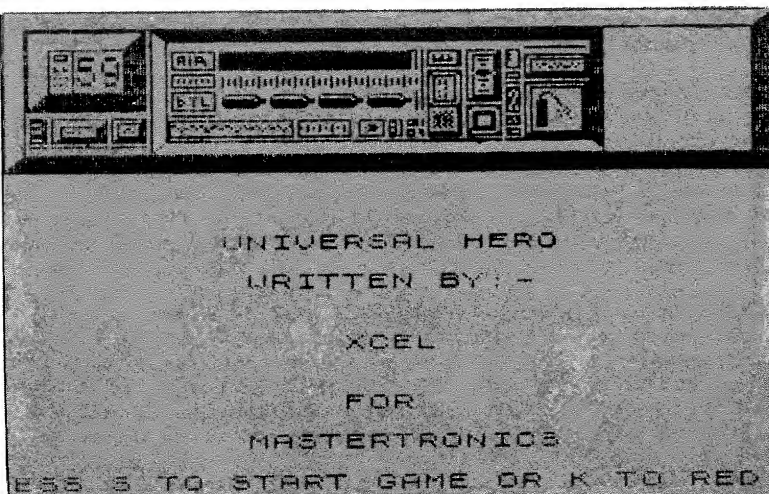
A feladat megoldása több fázisból tevődik össze. Először is a komphoz tartozó űrsiklónkkal le kell szállnunk egy kisebb holdon, ahol plutóniumot kell gyűjtenünk. Ezután egy másik bolygóra repülve ásványokat kell összeszednünk az űrhajó meghajtásához. Ezen a bolygón barlang is található, melyben egy öreg varázsló él. Űrsiklónkra csak az ő segítségével juthatunk vissza.

A játék színvonalas, ám van egy nagy hátránya, sajnos KEMPSTON típusú joystick Interface-szel nem játszható, a menüben csak a billentyűzet átdefiníálására van lehetőségünk. Ilyenkor be kell állítani a jobbra/balra/kurzor jobbra/kurzor balra/repülés/felvenni/használni/leírni funkciókat. Szerintünk a legcélszerűbb, ha az O,P,0,9,Q,A,M,Z billentyűkre állítjuk be az irányítást. A kurzor jobbra/balra funkció segítségével a nálunk lévő tárgyak között válogathatunk. Egyidőben összesen 9 tárgyat tarthatunk magunknál. A leírás funkció a felvett tárgy nevét közli velünk, valamint ahhoz hozzárendeli annak legjellemzőbb tulajdonságát is: pl. güggyő hal.



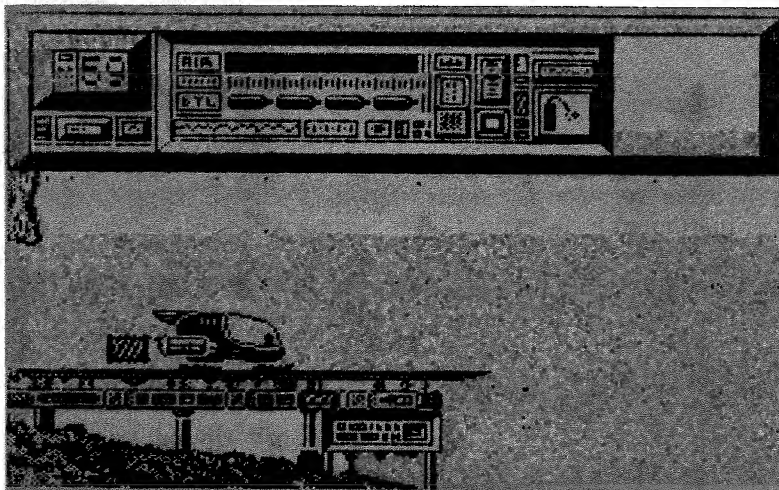
A rövidre sikerült bevezetőt követően indulunk el akciónkra. A mellékelt térkép START helyéről kezdődik a játék. Először is ballagjunk el jobbra és vegyük fel a csapat, majd ezt követően a megmunkálatlan űveget. Amikor elérkezünk a pálya szélére, egy nem lakható céljára szolgáló csövet látunk kilógni a falból. Ha most használjuk a csapat, meg-szüntetünk egy nagy nyomású csőtörést az alsóbb szinten. Űr-ruhánk sajnos már megjárta a hadak útját, ezért sérülékeny. Jól tesszük, ha kerüljük a találkozást minden gyanús egyénnel (az egyén fogalmába itt beletartoznak a robotok

is). Ereszkedjünk le a fal mellett, majd vegyük fel a három villás dugót és az űrsikló mellől a tekercs drótot. Visszafelé haladva, a messzelátó mellett ismét ereszkedjünk le, ahol a távkapcsoló hever. Ha ezt használjuk, kikapcsoljuk a floppy-disk-et őrző áramfalat. Balra megtaláljuk a személyazonossági lapot, ezt is vegyük fel, nagy szükségünk lesz rá. Most két szobával arrébb kóricálunk és a szimpatikusnak tűnő bűvár-dugattyút és a floppy-t is megkaparintjuk. Olyanok vagyunk mint a sárkányrepülők, a következő szobában ismét leereszkedünk, és elmegyünk balra, majd felfelé és lám csodák csodája a plutónium előállt. Van viszont egy kis gond, ha felvesszük a plutóniumot, elhalálunk, mivel az űr-ruhánk nem véd a radioaktív sugárzás ellen. Ezért ezt ott hagyjuk, és inkább lefelé, majd jobbra megyünk a rúd dinamitért, ezt követően vissza a leereszkedési helyre (azaz balra,fel,jobbra). Ha tovább megyünk jobbra a tűzépnel találjuk

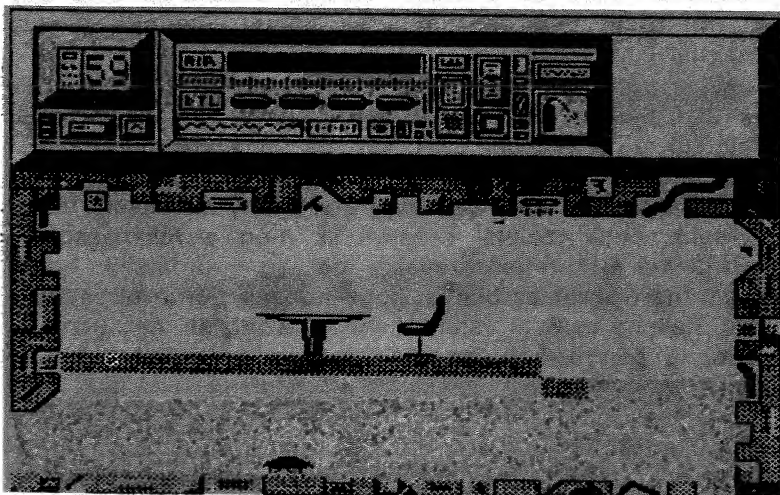


magunkat, ugyanis útkémben egy halom kő tornyosul. Most használjuk a dinamitot, a drótot, s végül a bűvár-dugattyút, melyek együttesen szétrobbantják az útkémben álló kőhalmot. 2 szobával tovább egy számítástechnikai kissezővetkezet leledzik, melyben a legfontosabb eszköz egy számítógép. A számítógépet nemes célokra használják, csak akkor hajlandó előtűnk megnyitni a falat, ha igazoljuk magunkat. Ehhez viszont először is be kell kapcsolnunk a számítógépet, amely a 3 villás dugó használatával lehetséges. A gép viszont azonnal ki fog kapcsolni, mindaddig, amíg a floppy lemezt nem használjuk. Ezután már kéri, hogy azonosítsuk magunkat. Gépünk be: 'SLARTIBARDFASTS'. Felvetődik a kérdés, vajon miért ezt gépünk be, a válasz egyszerű: mert ez áll a személyazonossági lapunkon. Ki fog nyitni az áramfal, sőt mi több még a szomszédos szobában is kinyílik egy titokzatos ajtó. Ez az ajtó nagy kincset, finomítatlan olajat rejt maga mögött. Ezt az olajat ősi arab szokás szerint kaparintsuk meg. Most keressük meg az olajfinomítót, amelyhez először menjünk le, és balra... Az olajfinomítóban kézenfekvő, hogy az olajat kell használnunk. Mivel a XXI. században vagyunk, a berendezés az olajból egyenesen az űrsiklóba alkalmas üzemanyagot állít elő, sőt most figyeljenek: még kannát is ad hozzá. Úgy gondoljuk a legcélszerűbb elmenni az űrsiklóhoz. Ha mindent pontosan végigkövettünk, nálunk kell, hogy legyen a megmunkálatlan üveg, az azonossági lap és az űrhajó üzemanyaga. Azt a tárgyat amit már egyszer használtunk, nyugodtan letehetjük, csak a helyet foglalja a kezünkben.

Az üzemanyagot használjuk az űrsiklónál. Fontos, hogy ha nem tudjuk igazolni magunkat, akkor nem repülhetünk, ez biztosítja, hogy illetéktelen személyek az űrsiklót ne használhassák (2053-ban megjelent 'Vagyonvédelmi Törvény' XIII. paragrafusa - egyébként még az antikváriumokban kapható). Az űrsikló most már elszállít minket az űrközpontba. Ahogy megérkezünk, rakjuk le a személyazonossági lapot és az üzemanyagot. A továbbiakban már nem hivatkozunk az útvonalra (ld. térkép), csak a tárgyak szerepét ismertetjük.



Felvehetjük az oxigén-palackot, majd ezt használva oxigén-tartályunk feltöltődik. Ha még van elég oxigénünk, célszerű ezt a palackot tovább vinnünk, későbbi feltöltés céljából. Vegyük fel a robotvisszahívót, majd repülünk le/jobbra/le - és a robotvisszahívó egységet használva visszahívunk egy droid-ot amely a radioaktivitás ellen védő dobozt őrzí. Szedjük fel a légycsapót, és a javító robotot. A megmunkálatlan üveget - amit a holdról hoztunk el -, a robotvisszahívó állomástól balra lévő szobában használjuk, s ennek következtében a csiszológép érzékeny lencsét csiszol belőle. Ezt is gyűjtsük be, majd menjünk fel, vegyük fel az útlevelet és a tojást. Ezt a tojást 2 szobával arrébb használhatjuk fel, és oxigén-palackunk feltöltődik. Itt a légycsapóval csapjuk fejbe a kukaradzsát, aminek eredményeképpen zsebünk gyűjteménye pozitívan alakul.

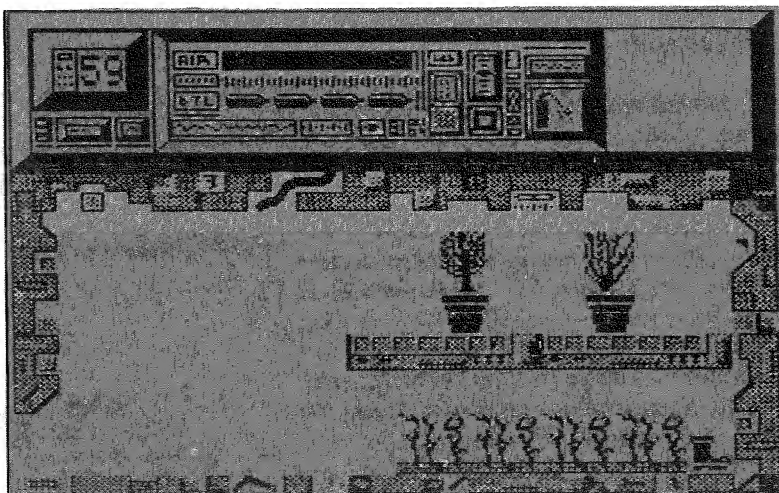


Az útlevelet vigyük az űrsiklóhoz. Az üzemanyaggal és a személyazonossággal visszajuthatunk a holdra (ha használjuk az útlevelet). Mindenesetre az indulás előtt győződjünk meg arról, hogy van-e legalább két üres hely a kezünkben. A holdon vegyük fel a radioaktivitástól védő dobozt, és most már felvehetjük a plutóniumot is. Visszamenve az űrhajóhoz, tegyük le a plutóniumot a 'minerals' felirathoz (egyébként ide az űrsiklótól a jobbra/le/jobbra/jobbra/jobbra/le/jobbra/le útvonalon juthatunk el). A radioaktivitás ellen védő dobozt ne tegyük le, ne-

hogy a végén előbb vegyük magunkhoz a plutóniumot.

Vegyük fel a tükröt, a 10 pennys-t, a fogantyú kezdetét, majd a barakk kulcsát is. Most pont kilenc tárgynak kell nálunk lennie: oxigén palack, javító robot, lencse, halott kukaradza, radioaktivitástól védő doboz, tükrő, 10 penny-s, fogantyú kezdete, valamint a barakk kulcsa. Most menjünk balra és használjuk a 10 pennys-t, ezáltal bepillantást nyerhetünk az űr rejtelseibe, de mivel semmi érdekeset nem látunk, menjünk jobbra/jobbra/fel és ismét jobbra. A húsevő növényeknél találjuk magunkat – érdemes elkerülni a velük való találkozást, mert a szép 'SLARTIBARDFASTS' név mellé megkaphatjuk a 'félkarú' jelzőt is. Szegény növények úgy látszik etetés előtt állnak, jól megéhezhettek a hosszú úton. Tálaljuk fel nekik az első-, és egyben utolsó fogást, a halott kukaradzsát. Furcsa dolog történik, a növények feldobják gyökereiket, ugyanis ilyen hosszú éhezés után a nehéz étel megárt a gyomornak. A következő szobában vegyük fel az öntözőkannát, és nyissuk ki a bezárt ajtót a kulccsal. Az ajtó mögötti szobában gyomirtót találunk, két szobával arrébb pedig gumicsizmát, amit elburjázott növények sora vesz körül. Nem kultúrált csak úgy a növények közé 'beslattyni', mert haragszanak, hogy régen öntözték meg őket. Megbékítésükre öntsünk rájuk egy kis gyomirtót.

Ezután a csizmát már meg-sze-rezhetjük, hozzuk is el, majd menjünk egy szobát balra, és az előző kárt helyrehozva használjuk az öntözőkannát. Most men-jünk vissza a csiszológéphez, majd a bal oldali szobában ve-gyünk fel a csavarhúzó-t. A követ-kező szobába bejuthatunk alulról és felülről is, kinek melyik a szimpatikusabb (a 'következő' szoba a csavarhúzó feletti). In-nen menjünk balra, majd ismét balra. A cikázó energianyalábot vezessük a falba, majd a lencsével ill. a tükrő segítségével fókuszáljuk a nyalábot az energiagyűjtő helyre. Ez fogja majd

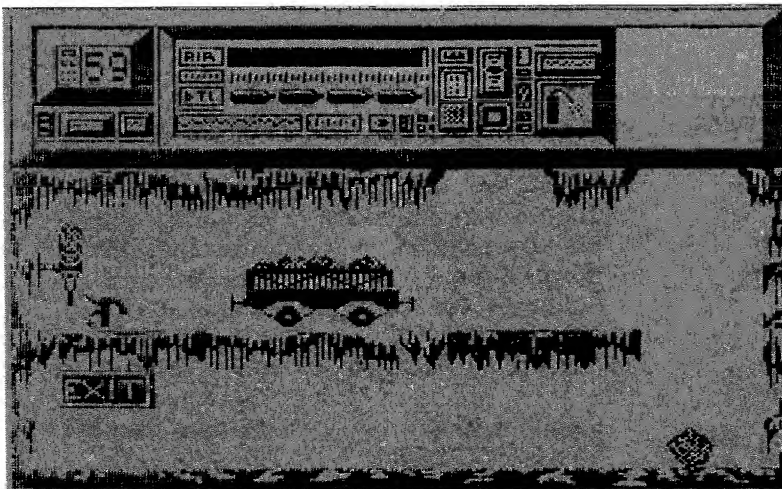


szolgáltatni a gépnek az energiát. Jobbra kijutva használjuk a csavarhúzó-t, majd a robotot, és a gép újra működésbe lép. Menjünk le, majd balra. Itt találjuk az űrkomp 'tele-transzportáló' készülékét, – ezt a gépet a személyzet csak vészhelyzetben szokta használni (akkor, ha a katasztrófa már elkerülhetetlen), ugyanis ez a bolygóra való lejutás utolsó lehetősége. A gépet használva viszont hatalmas energia, áram keletkezik, ezért kell magunkkal vinni a szigetelt gumicsizmát. Teleportálásunk csak akkor sikerül, ha a gép kabinjába állunk, egyébként sem sikerül használni a tárgyakat a szoba többi pontján. A gép úgy működik, hogy a tárgyat, vagy embert atomjaira bontja szét, és a megfelelő helyen egyesíti azokat.

A teleportálás után a bolygón találjuk magunkat, ahol is először ürítsük ki zsebeinket. Szedjük fel egy csövet, egy kevés kötelet, egy kést és egy nagy követ. Sajnos a barlang bejáratát egy újabb növényfajta őrzi, amitől úgy tudunk megszabadulni, ha használjuk a kést a bejáratnál jobbra lévő fánál. Hősrünk a kötéllel lehúzza a fát, ráköti a követ, és a késsel elvágja a kötelet. Eme ősi fegyverrel hatástalanítva lesz a növény, ami rövid találkozásunk után hamar kilehelte lelkét. Természetesen a kő is darabokra tört, most már könnyen bejuthatunk a barlangba.

Itt először vegyük fel a hajlott kulcsot, majd a pörölykalapácsot, aminek segítségével az üllőnél kiegyenesíthetjük a kulcsot. Ebben a pillanatban nálunk kell hogy legyen a kiegyenesített kulcs, a radioaktivitástól védő doboz és a cső. Gyűjtsük be a tengely-zsirt, vegyük fel a vízi-pumpát és ettől két szobával jobbra használjuk is, aminek következtében a barlangot elöntő vizet átszivattyúztuk a szomszéd szobába. Rakjuk le a pumpát és a csövet is a szegény hal mellé, majd menjünk jobbra. A varázsló ódjába jutottunk, ahol azonnal kaparintsuk meg a fékkapcsolót. Hamar rá kell jönnünk, hogy a varázsló számunkra érthetetlen szavakat motyog – az látszik, hogy nem a fékkapcsoló eltulajdonításáért mérges, mert akkor az biztos, hogy néhány perce piros fülű bülbül madárrá változtatott volna bennünket. Kellene keresni valami bennszülöttet, aki betöltené a tolmács szerepét. Ha kilépünk a barlangból, a gügyögő halon akad meg a szemünk. Hősrünk kapcsol: 'Ez jó lesz'.

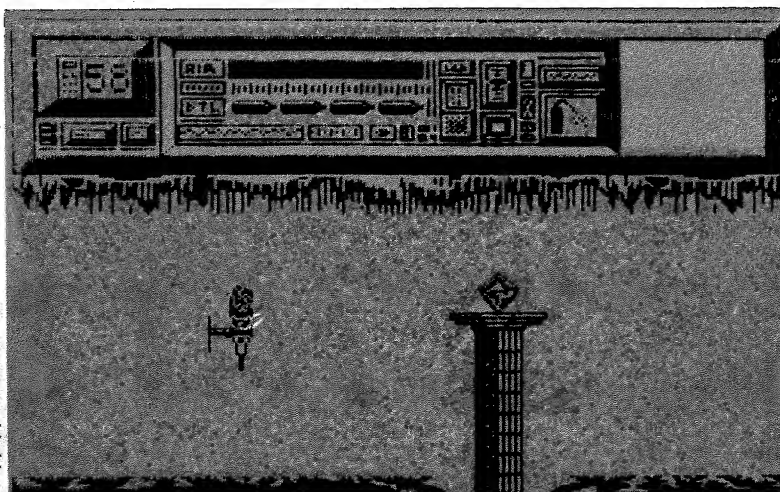
A gügyögő halat hónunk alá csapva caflatunk vissza a varázslóhoz. Most csak egy mondatot közöl velünk: 'Hozd vissza a gömböt!' Menjünk vissza arra a helyre, ahol a vízi-pumpát találtuk.



Most kell végrehajtanunk a játék egyik legveszélyesebb manőverét, a szoba legvégén használnunk kell a kiegyenesített kulcsot, viszont a szobát egy furcsa, elvarázssolt kő őrzi. Ez úgy néz ki, hogy ha a kő saját magá-
val egy szintben mozgást észlel, elindul, hogy elzárja a kijáratot. Ha a kő hozzánk ér, kellemetlen meglepetésben lesz részünk, ugyanis a 3 dimenziós testünk 2 dimenzióssá változik. Természetesen belátható, így könnyebb a közlekedés (pl. a padló és az ajtó közötti résen), de sajnos az oxigénpalackunk is összelapult, így a játék folytatására nincs lehetőségünk. Már indulás előtt

állítsuk a kurzort az egyenes kulcsra, és ereszkedjünk le a földre. Most gyorsan meg kell indulnunk az 'EXIT' (Kijárat) felirat felé. Ha eddig nem kapott el a kő, akkor már phhhúúú, megúsztuk. A szoba végén használjuk a kulcsot, így átjutunk a következő szobába. Ebben a szobában egy rozoga régi csille látható, használjuk a tengely-zsirt, majd a fékkapcsolót, s a csille két pályával arrébb szállít minket. Most össze kell gyűjtenünk az összes tárgyat, ami a barlangrendszerben található, így mielőtt a térkép által jelzett szögletes teleporthoz (t) mennénk, ellenőrizzük, hogy nálunk van-e minden.

Ha minden jól ment, nálunk van a rádióaktivitás ellen védő do-boz, a rubint, a kristály, a hatágú csillag, a gyűrű és a talizmán. A hatágú csillagot használva, egy zárt szobába, egy másik szögletes teleporthoz jutunk. Itt vegyük fel a kristálygömböt, majd a gyűrűt használva újra teleportáljunk, de most a varázslóhoz. A varázslónál használjuk a gömböt és 'Abrakad-abrak', az úrkompunknál találjuk magunkat, a t jelzésnél. Innen menjünk a 'mi-nerals' (ásványok) felirathoz, oda, ahova a plutóniumot tettük le. Vegyük fel a plutóniumot (ugye milyen jó, hogy nálunk van a rádióaktivitástól védő doboz), és használjuk. Az űrhajó gépei a nyersanyagokat üzemanyaggá alakítják át, a feladatot teljesítettük: 'Ügyes vagy, megakadályoztál egy nagy katasztrófát. X perced maradt...'



Sok sikert kívánunk a nagy küldetéshez!

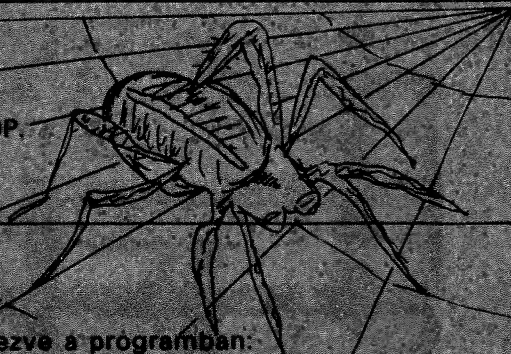
Agent X.

Az első szinten nem kell password.
A második szinten gépeljük be: HERE COME OL' FLAT TOP.
A harmadik szint kulcsa: THERE'S NO ESCAPING IT.

Super Soccer

Az örökélethez egy nagyon egyszerű 'CHEAT' lett elhelyezve a programban:

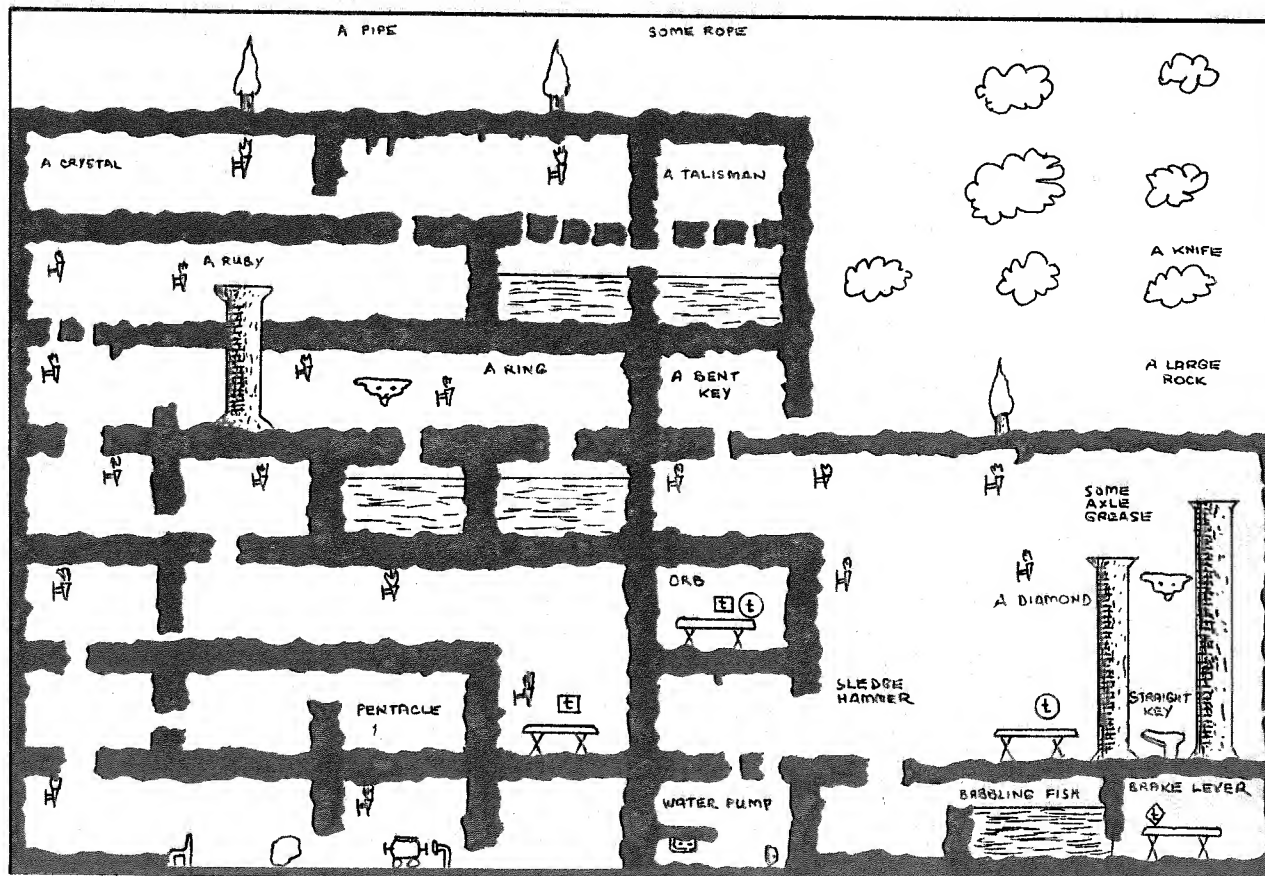
Nyomjuk meg egyszerre a T-'G'-H' billentyűket.



A Babbling Fish	-Egy gügyögő hal
A Bent Key	-Egy hajlított kulcs
A Crystal	-Egy kristály
A Diamond	-Egy gyémánt
A Knife	-Egy kés
A Large Rock	-Egy nagy kő
An Air Tank	-Egy légtároló
A Pipe	-Egy cső
A Ring	-Egy gyűrű
A Ruby	-Egy rubintkő
A Talisman	-Egy talizmán
A Tap	-Egy csap
A Stick of Dynamite	-Egy rúd dinamit
Brake Lever	-Fékkapcsoló
Curiosity	-Kíváncsiság
Dead Cocoroath	-Döglött csótány
Droid Recall	-Robot visszahívó
Egg	-Tojás
Floppy Disk	-Floppy lemez
Fly Swatter	-Légycsapó
Led Radiation Box	-Rádióaktivitás ellen védő doboz
Lens	-Lencse
Mirror	-Tükör
Orb	-Gömb

Sledgehammer
Some Axle Grease
Somecrude Oil
Some Rope
Spaceship Fuel
Spanner
Starting Handle
Straight Key
10 penny
Three Pin Plug
Two Core Wire
Watering Can
Water Pump
Weedkiller

- Útlevál
- Hatágú csillag
- Rózsaszín virág
- Búvár-dugattyú
- Plutónium
- A barakk kulcsa
- Távkapcsoló
- Javító Robot
- Durva üveg
- Gumicsizma
- Slartibardfasts
igazolványa
- Pörölykalapács
- Tengely-zsír
- Finomítatlan olaj
- Kevés kötél
- Űrhajó üzemanyag
- Csavarhúzó
- Fogantyú kezdete
- Kiegyenesített kulcs
- 10 penny-s érme
- 3 villás dugó
- Kéteres kábel
- Öntözőkanna
- Vizipumpa
- Gvomirtó



A KÉPERNYŐ NAGYÍTÁSA

Sok olyan észrevétel érkezett hozzánk az utóbbi időben, amelyben azt hiányolták, hogy kiadványunkban nem fektetünk hangsúlyt a BASIC nyelven megközelíthető trükkökre, a gépi kód ismeretét pedig nem feltételezhetjük általános szinten. Ebből kiindulva úgy döntöttünk, megpróbálunk olyan (minimális gépi kódú kiegészítésekkel ellátott) BASIC programokat ismertetni, amelyek – bár lassú futási eredménnyel –, de a gépi kódú rutinok színvonalán érnek el látványos eredményeket.

Ebben a részben egy olyan programot ismertetünk, amely a képernyőn megjelenített SCREEN meghatározott területét (kijelölt ablakot) a kétszeresére nagyítja. A kalkuláció végrehajtása BASIC-ben nem a leggyorsabb, de a program használatát kezdők is minden problémára nélkül eljáratíthatják.

```

10 CLEAR 50000
20 FOR i=65000 TO 65089: READ a: POKE i,a:
  NEXT i
30 PRINT AT 10,7; FLASH 1;"LOAD SCREEN$"
40 PRINT TAB 5;"Nyomd meg az 5-8 billentyű
  ket az ablak mozgatására, és az 'A' billen
  tyűt az ablak tartalmának kinagyítására"
50 LOAD""SCREEN$
60 PRINT #0;AT 0,0;"Nyomj meg egy billen
  tyűt!":PAUSE 0
70 PRINT #0;AT 0,0;"
  "
80 RANDOMIZE USR 65000: CLS
90 LET a=0: LET b=175
100 IF INKEY$="a" THEN GO TO 180
110 IF INKEY$="5" THEN LET a=a-8: IF a<=0
  THEN LET a=0
120 IF INKEY$="6" THEN LET b=b-8: IF b<=87
  THEN LET b=87
130 IF INKEY$="7" THEN LET b=b+8: IF b>=175
  THEN LET b=175
140 IF INKEY$="8" THEN LET a=a+8: IF a>=128
  THEN LET a=128
150 RANDOMIZE USR 65012: PLOT OVER 1;a,b:
  DRAW OVER 1;127,0: DRAW OVER 1;0,-87:
  DRAW OVER 1;-127,0: DRAW OVER 1;0,86
160 PAUSE 0
170 GO TO 100
180 LET bb=(ABS (b-175))/8*32: LET aa=INT
  a/8: LET p=16384: LET a=a+1
190 FOR s=1 TO 11
200 LET x=(ABS (b-175))/8
210 IF x>=0 AND x<8 THEN LET z=x*32+
  57968+a/8-256+32
220 IF x>=8 AND x<16 THEN LET z=(x-8)*32+
  60016+a/8-256+32

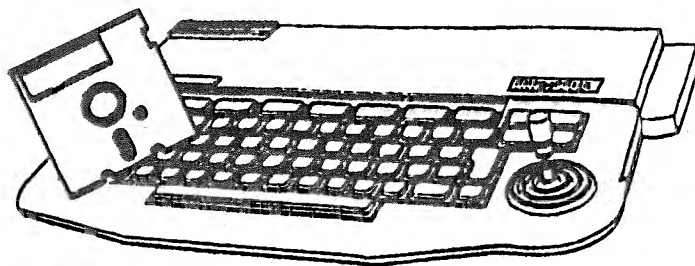
```

```

230 IF x>=16 THEN LET z=(x-16)*32+62064+
  a/8-256+32
240 IF s=5 THEN LET p=18432
250 IF s=9 THEN LET p=20480
260 FOR i=0 TO 1792 STEP 256
270 LET z=z+256
280 FOR j=p TO p+31 STEP 2
290 LET w=z+(j-p)/2: POKE 64920,w-256*INT
  (w/256): POKE 64921,INT (w/256)
300 RANDOMIZE USR 65025: POKE j, PEEK
  64922: POKE j+1,PEEK 64923: POKE j+256,
  PEEK 64922: POKE j+257,PEEK 64923
310 NEXT j
320 LET p=j+480: IF i=768 THEN LET p=p-2016
330 NEXT i
340 LET p=p-2016: LET b=b-8
350 NEXT s
360 FOR f=22528 TO 23231 STEP 64
370 FOR t=0 TO 31 STEP 2
380 LET i=PEEK (64144+bb+aa): POKE f+t,i:
  POKE f+t+1,i: POKE f+32+t,i: POKE f+33+t,i
390 LET aa=aa+1
400 NEXT t
410 LET aa=aa+16
420 NEXT f
430 INPUT "Továbbbi nagyítás? I/N";a$: IF a$
  (TO 1)="I" THEN GO TO 80
440 INPUT "Régi vagy új kép? R/U";a$: IF a$
  (TO 1)="R" THEN CLS: GO TO 90
450 GO TO 30
460 DATA 1,191,26,17,144,226,33,0,64,237,176,
  201,1,0,24,17,0,64,33,144,226,237,176,201,0,
  237, 91,152,253,26,71,62,0,203,120,40,2,198,
  192,203, 112,40,2, 198,48, 203,104,40,2,198,
  12,203, 96,40,2,198,3,50,154,253,62,0,203,88,
  40,2,198,192, 203,80,40,2,198,48,203,72,40,2,
  198,12,203,64,40,2,198,3,50,155,253,201

```

RUN kiadását követően rövid üzenet jelenik meg, és a program várja egy **SCREEN\$** típusú file betöltését. Keressünk egy tetszőleges képernyőt (akár saját, akár jártékprogramból krakkolt), és töltsük be. Betöltést követően nyomjunk meg egy billentyűt, és megjelenik egy 10x15 karakter méretű ablak. Ezt mozgathatjuk az 5-8 billentyűk segítségével. Állítsuk be a megfelelő – nagyításra szánt – képrészletet, és nyomjuk meg az 'A' billentyűt. Rövid idejű kalkulációt követően megjelenik a nagyított képrészlet. Érdekessége a programnak, hogy az attribútumok is a helyükre kerülnek, így színhe-lyes nagyítást kapunk eredményül. A nagyítást követően választhatunk, hogy ezen a képen újabb nagyítást akarunk-e végrehajtani, ill. kérjük-e a régi képet, vagy egy újat töltsünk be kazettás magnetofonról. Az algoritmus első látásra bonyolultnak tűnik, de ha sorról-sorra áttekin- tük a programot, hamar eljáratíthatjuk mechanizmusát.



Hova POKE-oljunk?

A 8.részben közöltük a 'Nodes of Yesod' c. játék öröklétesítéséhez szükséges adatokat. A levelek tanulsága szerint ezt azonban kevesen tudták használni. Most megpróbálunk egy részletesebb tájékoztatóval szolgálni a nemes cél elérése végett.

A játékprogramok megjelenése óta léteznek cracker-ek is, ők azok akik feltörik a játékokat. Tevékenységük szerint három csoportra oszthatók:

- másolhatóvá teszik a programokat
- megkönnyítik a játék menetét (öröklés, végtelen energia, stb.)
- rutinokat gyűjtenek a játékokból

A legártatlanabb krakkolás (öröklétesítés) sajnos a legtöbb esetben igen nehéz, mert ahhoz, hogy meg tudjuk keresni a programban az ehhez szükséges részeket, előbb ártatlanná kell tenni a játék másolása ellen elhelyezett bombákat. Ez a védelemről és a programozóktól függően 1 óra és két hét közötti munkát jelenthet a krakkolóknak. Miután a védelmet lerobbantottuk, elkezdhetünk turkálni a kódban, megkereshetjük az életék levezetésért felelős rutint. Megjegyzendő továbbá az a tény is, hogy egy ilyen vállalkozás előtt nem árt tisztában lenni az ENTERFACE felépítésével, valamint a BASIC programozási nyelvről le kell mondani, és a gépi kódot kell előnyben részesíteni. Határozottan érdemes felfegyverkezni különböző segédprogramokkal (monitor, debugger, tracer, stb.) nem árt, ha ilyen esetekben akad Elanum is a gép közelében.

Most konkrét példaként tekintsük meg a 'Nodes of Yesod' esetét.

JÓ HÍR! JÓ HÍR! JÓ HÍR!

A program nem lett ellátva másolás elleni védelemmel!!! Ráadásul még egy blokkban is töltődik be, mi több EXOS 5-ös fejléccel...

Lássuk, hogy is van ez: a program egy adat blokkból áll, amely A7D0h hosszú. A blokkban, a program előtt található egy 10h hosszú un. EXOS fejléc is, amely meghatározza a program

betöltésének módját. A fejléc felépítése a következő:

- 00 - nem használt
- 05 - típus byte
- xx - program hossz (low byte)
- xx - program hossz (high byte)
- 00 - a fejléc többi 12 byte-ja zérus

Az 5-ös típusú fejléc tulajdonképpen azt jelenti, hogy az EXOS magát a programot 100h-ra betölti, majd átadja a vezérlést a 100h-ra. A leközölt módosítások címhelyesek voltak, ezért úgy a legkönnyebb bevinni őket, hogy az egész blokkot behúzzuk 00F0h-ra, az adott címeken átírjuk az adott byte-okat, majd kimentjük a blokkot 00F0h-tól A7D0h hosszban. Amennyiben a monitorunk nem teszi lehetővé a 00F0h-ra történő töltést, adjuk hozzá a címekhez a betöltési offset-et is. Azok számára, akik nem járatosak a gépi kódú programozásban, a következő megoldást ajánljuk. BASIC-ben írunk egy rövidke programot, amellyel byte-onként beolvassuk az eredeti programot, majd byte-onként kiírjuk az új programot, csak a megfelelő byte-okat cseréljük le.

```
100 OPEN#106:"TAPE1:file nev" ACCESS INPUT
110 OPEN#107:"TAPE2: uj file nev" ACCESS
    OUTPUT
120 FOR =0 TO 42960
130 GET#106:A$
140 IF A=856-16 THEN LET A$=CHR$(0)
150 IF A=857-16 THEN LET A$=CHR$(0)
160 IF A=858-16 THEN LET A$=CHR$(0)
170 IF A=859-16 THEN LET A$=CHR$(0)
180 IF A=908-16 THEN LET A$=CHR$(0)
190 PRINT#107:A$;
200 NEXT A
210 CLOSE#106
220 CLOSE#107
```

Ha magnetofont használunk, érdemes kettővel dolgozni, így is el lesz vele a gép egy darabig, de utána már nem lesz problémaavigiátszani és megnyerni ezt a rendkívül szimpatikus játékot.

ENTERFACE

BEACH-HEAD™

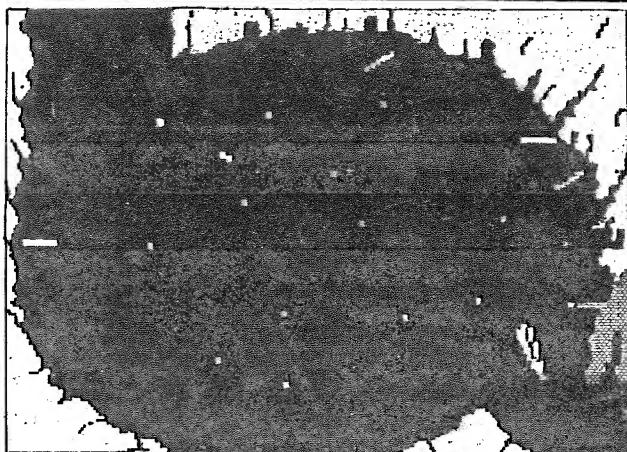
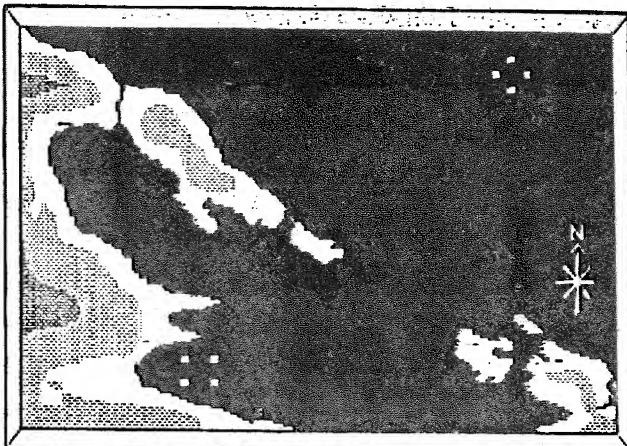
Bizonyára minden ENTERPRISE felhasználó hallott már erről a játékról, főképpen akkor, ha a közelében ismerőse rendelkezik C64 vagy Spectrum számítógéppel. A maga idejében színvonalas játék most már az ENTERPRISE gépet is meghódította, reméljük nem hiába.

Mint tudjuk, 1944 június 6.-án az angol-amerikai csapatok Normandiában partraszállást hajtottak végre. Ezt a hadműveletet szimulálja a játék úgy, hogy a csatában a szövetséges csapatok oldalán veszünk részt.

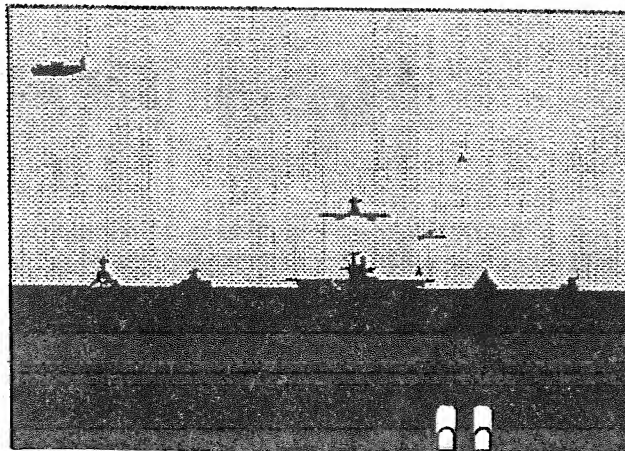
A program a szokásos módon tölthető be, betöltés után bejelentkezik a címkép, amin látható a programot készítő cég emblémája, és megjelenik a 'Press key to menu' üzenet, vagyis 'Nyomj le egy billentyűt a menü előhívásához'. Ha ezt elmulasztjuk, pár másodperc elteltével demonstrációt láthatunk a játékról, amelyben megtekinthetjük az egyes szinteket. Tetszőleges billentyű megnyomása után bekerülünk a menübe:

S - Start: a játék indítása / J - Select Joystick: joystick választás / I - Instructions: Információt kérhetünk a játékról / P - No. of Players: Beállíthatjuk a játékosok számát / K - Define keys: Billentyűzet átdefiníálása / L - Skill Level: Itt állíthatjuk be a játék erősségét;

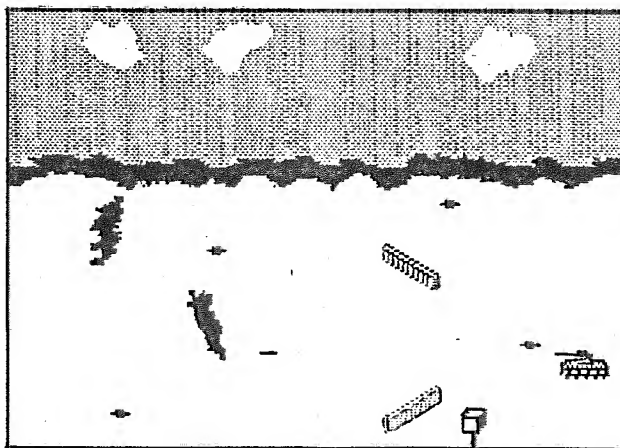
Először egy partrészlet sematikus rajzát látjuk. A villogó célkereszt jelzi pozíciókat. Ha a tengerszoros felé vesszük útunkat, bejutunk egy aknákkal telítődött öbölbe, itt az a cél, hogy minél több hajót sértetlenül átvigyük az öböl túlsó oldalára.



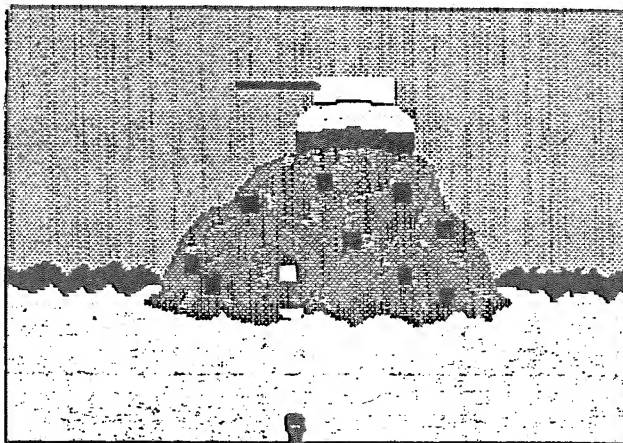
A következő lépésben pozíciókat egyeztetjük az ellenséges flotta pozíciójával, és bekerülünk a játék legjobban megrajzolt, s tökéletes hanghatásokkal telítődött pályájára, tengeri csatát vívhatunk. Itt először gépágyúnkkal szórni kell az ellenséges vadász- és szállító gépeket, majd bizonyos pontszám elérése után ágyúznunk kell az ellenséges hajókat.



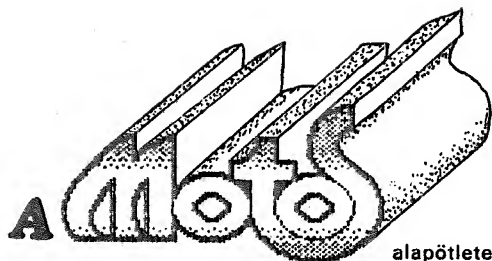
Ha minden hajót elsüllyesztettünk, a célkeresztet a parthoz kell igazítanunk, és további hadműveletünket egy tank segítségével visszük végbe tankcsapdák, aknák, zártóüz összehatásában.



A csatát akkor tudjuk megnyerni, ha az ellenség kulcsfontosságú bázisát megsemmisítjük. Ehhez be kell lőnünk az aktuális, világító nyílásokon. Sietnünk kell, mert az ellenség is észlelt bennünket, s ha nem vagyunk elég gyorsak, kezdetjük újból az egészet.



A BEACH HEAD jó grafikája és hanghatásai alapján jogosan sorolható a legjobb ENTERPRISE játékok közé. Jó szórakozást, és sikeres partraszállást kívánunk!



alapötlete a NAMCO

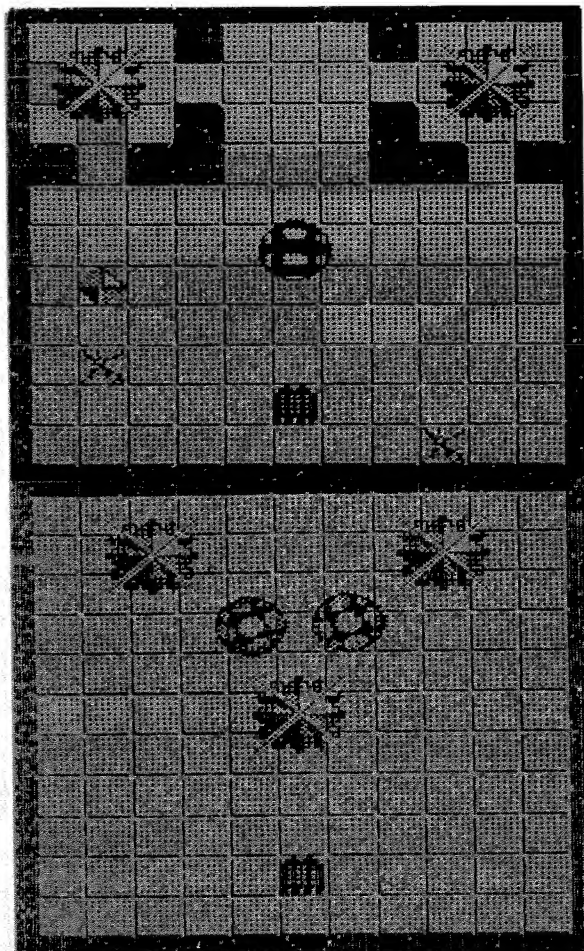
Ltd. nevéhez fűződik, ők 1985-ben dobták piacra ezt az ügyességi játékot, melynek ENTERPRISE változata megegyezik a Spectrum változattal (Mastertronic 1987).

Az ENTERPRISE változat készítőjét(-it) az ismeretlenség homálya fedi, mivel a program Screen-jében nem mutatkozik(-nak) be, és az eredeti gyártót, a forgalmazót, valamint a gyártási évet lefedték.

Ezen túlmenően a 2-es billentyűvel választható MONOCHROM - MULTICHROM (Fekete/fehér - Színes) lehetőség is elmaradt, bár a menüben mint lehetőség szerepel, csak éppen nem működik. A program tehát állandóan színes üzemmódban van, bár ez fekete/fehér TV-k ill. MONOCHROM monitorok felhasználása esetén nem okoz gondot.

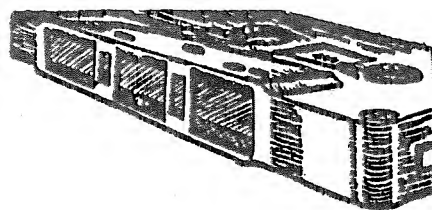
A játékot ketten is játszhatják, a választás az '1'-es billentyű segítségével lehetséges.

Egy 12x12 négyzetből álló mezőben, illetve ennek különböző változataiban játszunk:



Célunk, hogy a játékos által irányított űrhajó figurával (MOTOS) az ellenfeleket a játéktérrel leledőssük. A programozók gondoskodtak arról, hogy a művelet ne legyen túl egyszerű, így az ügyességen kívül elő kell venni a kis agyunkat is.

Kezdetben öt élettel rendelkezünk, melyet a képernyő jobb alsó sarkában látható űrhajó figurák jeleznek. Ezen a részen látható két négyzet is, bennük 'NORMAL' felirattal. A bal oldali négyzet a lököerőt, míg a jobb oldali négyzet az ugróerőt jelzi. A játék során az ezeknek megfelelő ábrákat kell összegyűjteni, melyek a megfelelő négyzet felett megjelennek. Ha valamiből többet gyűjtünk



össze, mint amit képes a program kijelezni, akkor a gép nem 'nyeli le', hanem egy kis számmal jelzi, hogy mennyi van még a kijelzett erőből.

Ha elhalálozunk, és ez bizony gyakran elő fog fordulni, akkor a gép visszaadja a már beválasztott erőt, vagy újrást, és újabb választásra ad lehetőséget. Mivel az eredeti SPECTRUM program kifogástalanul működik az EMULATOR kártyával is, és az öröklet: POKE 42241,0 a loader-en keresztül gond nélkül bevihető (ld. SpV 9.rész), így az előbb említett tulajdonságot fel tudjuk használni arra, hogy energiánkkal takarékoskodjunk.

Ha egy olyan helyen vagyunk, ahol több egység lököenergiát igénylő ellenfél mellett gyengébbek is vannak, akkor a következőképpen járunk el:

Válasszuk ki a megfelelő mennyiségű lököenergiát, szorozzuk az erősekkel addig, amíg a gyengék elő nem kerülnek a fészkekből. Ezután MOTOS-zkáljuk le az erőseket, majd magunk is kövessük őket. Most a gép újból kéri az energia, stb. választást, de mi nem választunk semmit, hanem leledőssük a megmaradt gyengéket, és energia-vesztés nélkül mehetünk a következő pályára. Ha ugrunk, akkor a leérkezésünk helye megrepedezik. Ha még egyszer ide tottanyunk, akkor úgy eltávoznak az örök vadászmezőkre, mintha a pálya széléről zúgtunk volna le.

A programozó bácsik úgy gondolták, hogy még így is túl egyszerű, ezért egy kis idő elteltével meteorit-eső zúdul le a pályára, csinos ki lyukakat vájva a terepen, amiken szintén lepotyoghatunk. Valamennyire elégtételt ad az, hogy a 'csúnyák' is úgy járhatnak mint mi.

Még egy kis kedvesség a programozóktól, hogy az első kavics általában oda érkezik, ahol éppen mi vagyunk. Amennyiben úgy hozza a helyzet, hogy elfogyott a lököerőnk, és az ellenfelünk erősebb, két lehetőségünk van:

- Megvárjuk a meteorit-záport, és ha ügyesek vagyunk, akkor az ellenfelünket előbb éri találat, mint minket.

- Ha van ugróerőnk, akkor az ellenfelünk felett átugrálva a padlót belyukasztjuk és az ellenfelet 'lyukra csaljuk'.

Az ellenfelek leledéséhez a játék kezdetén az alaperő is elég. A pályák számának emelkedésével azonban a leledéshez egyre több erő kell. Pl. a ROBOT-hoz 3-4 egység, a LABDA-hoz 3-5 egység.

Az örökletes változatnak egyébként van egy szépséghibája:

A JÁTÉKOT NEM LEHET BEFEJEZNI!!!

Tudniillik az utolsó pálya után újra az utolsó pálya következik, így csak a legjobb gomb segít (RESET), ha van, vagy az áramtalanítás.

Végül ismertetjük a játék során alkalmazott jelölések jelentését:

	Labda		Pók
	Fóka		Lepke
	Rák		Robot
	MOTOS		Golyó-fészkek
	Gyémánt 1000 pont		Golyó
	Lökőerő 1 egység		Tűz-golyó
	Ugróerő 1 egység		Gomb
	Sérült padló		



1. SZÍNSKÁLA ANIMÁCIÓ

```

110 DIM CLR(9,8)
120 FOR X=0 TO 8
130   FOR Y=1 TO 7
140     READ CLR(X,Y)
150   NEXT X
160 NEXT Y
170 SET 22,1
180 SET 23,2
190 SET 25,27
200 OPEN #101:"video:"
210 DISPLAY #101:AT 1 FROM 1 TO 27
220 LET INK=0:LET CN=0:LET CN1=2
240 FOR X=600 TO 700 STEP 3
250   LET CN=CN+1
260   IF CN>CN1 THEN LET CN=0:
       LET CN1=CN1*1.2:CALL INKINC
270   PLOT 0,X:1279,X
280 NEXT X
370 PLOT 640,260,
380 FOR X=1 TO 200 STEP 20
390   CALL INKINC
400   PLOT ELLIPSE X,200-X,
410 NEXT X
425 CALL PALSWITCH
430 DEF PALSWITCH
440   LET Z=2
450   DO
460     IF Z=1 THEN
470       LET Z=2
480     ELSE
490       LET Z=1
500     END IF
510     FOR X=0 TO 7
520       IF Z=1 THEN LET A=CLR(X,1)
530       IF Z=2 THEN LET A=0
540       LET B=CLR(X,2):LET C=CLR(X,3):
           LET D=CLR(X,4):LET E=CLR(X,5):
           LET F=CLR(X,6):LET G=CLR(X,7)
550       FOR Y=1 TO 20
560         SET PALETTE 0,G,A,B,C,D,E,F
570         SET PALETTE 0,F,G,A,B,C,D,E
580         SET PALETTE 0,E,F,G,A,B,C,D
590         SET PALETTE 0,D,E,F,G,A,B,C
600         SET PALETTE 0,C,D,E,F,G,A,B
610         SET PALETTE 0,B,C,D,E,F,G,A
620         SET PALETTE 0,A,B,C,D,E,F,G
630       NEXT Y
640     NEXT X
650     RESTORE
660   LOOP
670 END DEF
690 DEF INKINC
700   LET INK=INK+1
710   IF INK>7 THEN LET INK=7
720   SET INK INK
730 END DEF
740 DATA 64,8,72,1,65,9,73
750 DATA 64,136,200,17,81,153,217
760 DATA 128,80,208,10,138,90,218
770 DATA 128,16,144,2,130,18,146
780 DATA 192,24,216,3,195,27,219
790 DATA 128,48,176,6,134,54,182
800 DATA 0,32,32,4,4,36,36
810 DATA 64,40,104,5,69,45,109
820 DATA 192,56,248,7,199,63,255

```

2. 3D GRAFIKUS ÁBRA

```

100 SET STATUS OFF
110 DIM P(2,20),V(2,20)
120 LET CHOICE=1
130 LET C=10
140 SET 22,1
150 SET 23,1
160 SET 27,RED
170 OPEN #C:"VIDEO:"
180 SET #C:PALETTE RED,YELLOW,BLACK
190 SET #C:INK 1
200 DISPLAY #C:AT 1 FROM 1 TO 24
205 PLOT #C:50,40
210 LET XX=1
220 LET T=XX
230 FOR X=1 TO 2
240   FOR Y=1 TO 20
250     ON CHOICE GOSUB 410,420
260     LET P(X,Y)=INT((XX/2+Y/2)*60)
270     LET V(X,Y)=INT((Z+XX/2-Y/2)*40+400)
280   NEXT
290   LET XX=XX+1
300 NEXT
310 LET X=1
320 FOR Y=1 TO 19
330   PLOT #C:P(X,Y),V(X,Y);P(X+1,Y),V(X+1,Y);
       P(X+1,Y+1),V(X+1,Y+1);P(X,Y+1),V(X,Y+1);P(X,Y),V(X,Y)
340 NEXT
350 LET T=T+1
360 LET XX=T
370 IF XX=19 THEN 430
380 GOTO 230
390 LET CHOICE=2
400 GOTO 210
410 LET A=XX-10:LET B=Y-10:LET Z=
       COS(SQR(A*A+B*B))*2:RETURN
420 LET A=XX-10:LET B=Y-10:LET Z=
       EXP(6-(SQR(A*A+B*B)))/35:RETURN
430 SET #C:INK 2
440 PLOT #C:1,1,PAINT
450 LET C=11
460 SET 27,BLACK
480 IF XX=19 AND CHOICE=2 THEN 540
485 IF CHOICE=2 THEN 540
490 LET CHOICE=2
500 GOTO 140
540 LET C=10
560 DO
570   IF JOY(0)=8 THEN LET C=10
580   IF JOY(0)=4 THEN LET C=11
590   DISPLAY #C:AT 1 FROM 1 TO 24
600 LOOP

```

Az első demonstrációs program a teljes palettán végigfutó színskála árnyalatok bemutatásával szemlélteti a sprite-eltűntetés egy igen egyszerű és látványos módszerét. A második program térbeli grafikus alakzat megjelenítését segíti elő.

4 PRINT'N'PLOTTER cég által forgalmazott **SCREEN MACHINE** a 48K Spectrumra készült igen nagy értékű, hatékony tervező eszköz. Egyéb software-rel (pl. **PAINTBOX**) kombinálva, mind kezdők, mind haladók számára megszámlálhatatlanul plusz szolgáltatást nyújt. Régóta vártak már a programozók olyan programra, amely támogatja a gyorsan hozzáférhető, és rugalmasan kezelhető gépi kódú rutinokat, amelyek hasznosak lehetnek akár **BASIC**-ből, akár gépi kódú oldalról elérhető rajzprogramok számára. A **SCREEN MACHINE** segítségével a grafikus programozás egyszerűbbé válik, használatával lehetőség nyílik a grafikus képernyők tömörítésére, memóriatakarékos szövegszerkesztésre, több képből álló képállományok létrehozására, az állományok gyors cseréjére, a grafikus kép egyes részleteinek nagyítására, kicsinyítésére, a grafikus kép átszínezésére, függőleges tengelyre történő tükrözésére, mozgatására, ill. annak magnetofonra, vagy microdrive-ra történő rögzítésére is.

A SCREEN MACHINE BEHÍVÁSA

A program egy index betöltőt, három önálló programot, és egy demo-t tartalmaz. File térképe sorban a következő: 1127//3526/2989/2390//1189/5000/2713//1700//1648/3175/37858. A **LOAD**"" utasítással az 'index' nevű betöltő file jelentkezik be. Itt meg kell állítanunk a magnetofont, s a képernyőn megjelenik a főmenü:

1. **SCREEN MACHINE ONE:**
képtömörítő (töltési idő: 43 mp.)
2. **SCREEN MACHINE TWO:**
grafikaprocesszor (tölt.idő: 48 mp.)
3. **STRING MACHINE:**
szövegszerkesztő (töltési idő: 40 mp.)

Az 1-3 billentyűkkel kiválaszthatjuk a megfelelő állományt, majd a magnetofon elindítása után a megfelelő rész betöltődik.

Megjegyzés: Az egyes rész-programok külön-külön is behívhatók.

Ebben a részben az első résszel, a képtömörítővel fogunk megismerkedni.

SCREEN MACHINE ONE (SCREEN COMPRESSOR)

Ez a program segédeszközünk lesz akkor, amikor grafikát akarunk tömöríteni, képekből álló adatállományt akarunk létesíteni, vagy éppen váltott képekből álló kép-animációt tervezünk. Azt bizonyára mindenki tudja, hogy a Spect-

rum 48K gép memóriájába hat teljes (6912 byte hosszú) képernyő fér. Gondot okoz ekkor az, hogy mindössze 308 szabad byte-unk áll rendelkezésre **BASIC** program írásához. Ha megelégszünk 5 teljes képpel, úgy már 7 kbyte szabad felhasználói területünk marad. Programok ill. képek tárolása esetén több módszerrel is megtakaríthatunk jelentősebb memóriaterületet:

- Nem használjuk csak az egyik képernyő harmadot, így képeink számát 5-ről 15-re növelhetjük.
- Ha a képünk egyszínű, akkor nincs szükség az **ATTRIBUTUM** memóriára, teljes kép esetén 768, képernyő-harmad esetén 256 byte-ot spórolhatunk.
- A leghatékonyabb eljárás a memória tömörített formájú tárolása, ilyenkor betöltést követően alakítjuk vissza a memóriát eredeti állapotába.

A **SCREEN MACHINE ONE** lehetővé teszi az itt említett három eljárás-mód tetszőleges kombinációban történő alkalmazását.

A KÉPTÖMÖRÍTŐ

A legtöbb **SCREEN\$** állományú file-ban vannak üres területek, vagyis zérus byte-ok. A **SCREEN MACHINE ONE** megszámlálja ezeket, és számukat egyetlen adatként rögzíti. Visszahíváskor helyükre kerülnek a tömörített byte-ok, és előtűnik az eredeti kép.

A tömörítés hatékonysága nagyban függ a kép kitöltött és üres helyeinek arányától. A memóriát a legtakarékosabban az egyszínű vonalas ábrák, és képharmadok használják ki, amelyekben nincs nagyobb összefüggő **INK** rész. A **SCREEN MACHINE** demo-jában látható zebra kép egy attributum nélküli teljes kép, de 6144 helyett 2475 byte-ban elfér, azaz 4 kbyte megtakarítást tesz lehetővé.

Mivel számolni kell a zérus byte-okat, a tömörített grafika megjelenítése a szokottnál lassabb lesz, ez azonban - kivéve a gyors mozgású animációkat - még mindig megfelelő sebességű, nem beszélve arról, ha a **BASIC**-kel hasonlítjuk össze.

Az alapmenü:

- | | |
|--------------|-------------------|
| 1. Catalogue | (katalógus) |
| 2. Store | (behívás/tárolás) |
| 3. Load | (behívás) |
| 4. Save | (kimentés) |

Az egyes menüpontokat tekintsük át a felhasználás valószínű sorrendjében.

Első teendők egyértelműen egy - már kész - grafika behívása lesz, abból a célból, hogy tömörítsük, majd tömörített formában kimentsük. Válasszuk ebből adódóan először a 2-es opciót (store) és hívjuk be az előzőleg kazettára mentett screen-t. Fontos megjegyezni, hogy ilyenkor csak SCREEN\$ állományú file-ok behívására van lehetőségünk. A PAINTBOX által előállított gépi kódú screen-file-ok itt nem tölthetők be. Amikor a screen hibátlanul betöltődött, kint marad a képernyőn, és két újabb lehetőségünk adódik:

C: működésmód megváltoztatása (CHANGE)

K: működésmód megtartása (KEEP)

Ezen túl kijelződik a képernyőn az is, hogy a kép milyen üzemmódban került tárolásra. Az aktuális (érvényben lévő) működésmód attól függ, hogy a kép előzőleg milyen módon került kimentésre. A következőkben egy táblázatot közlünk, amely felsorolja a lehetséges működésmódokat:

	Norm +attr	Norm -attr	Tömör +attr	Tömör -attr
Teljes kép	3	2	35	34
Felső 1/3	5	4	37	36
Középső 1/3	9	8	41	40
Alsó 1/3	17	16	49	48

A 3. üzemmód (MODE-3) tehát a megszokott, teljes képernyő, attribútumaival együtt. Hossza 6913 byte, azért hosszabb 1 byte-tal a megszokottnál, mert a visszahívó rutinak ismernie kell az üzemmód-számot az azonosításhoz.

Amennyiben az aktuális üzemmódot meg akarjuk tartani, a 'K' billentyűt kell megnyomnunk, ezután újabb opciók állnak rendelkezésünkre:

C: megadja azt a byte-számot, amennyit a fennálló üzemmód igényel (COUNT)

S: a rendelkezésre álló legalacsonyabb memóriarekesztől kezdve elhelyezi a képet a memóriában (STORE). Vigyázzunk, ne kíséreljünk meg elhelyezni képet a memóriában, ha az már nem fér el.

M: visszatérés az alapmenühöz

STORE választása esetén a kép kikerül a memóriába, és megjelenik a katalógus-lap. Ne felejtjük el, hogy ez mindenkor meghívható az alapmenüből is az '1' opcióval.

A CATALOGUE feljegyzí minden eltárolásra került kép memória-helyzetét, hosszát, tárolási üzemmódját, valamint két azonosító számot is rögzít, amelyeket a visszairatáshoz POKE segítségével a vezérlésnek majd meg kell adnunk. A screen file-ok tárolására rendelkez-

zésre álló memóriaterületünk 35235 byte hosszú, amelyhez elméletileg 72 kép (number of SCREEN\$) tartozik. Ez az érték egyébként valószínűtlen, mert ekkor egy-egy kép max. 489 byte hosszú lenne. Ha a még rendelkezésre álló szabad memóriaterület nagysága kevesebb mint 6920 byte, a képernyőn egy villogó 'MEMORY' figyelmeztetés jelenik meg. Menet közben ezért feltétlenül használjuk a 'COUNT' funkciót, mielőtt kiadnánk a soron következő 'STORE' utasítást.

Feltételezzük, hogy memóriatakarékossági okokból ki akarunk szállni a fennálló üzemmódból, ill. pl. a képernyőnek csak egy bizonyos részét kívánjuk eltárolni, ilyenkor a 'K' opció (KEEP) helyett a 'C'-t (CHANGE) választjuk ki. Ennek hatására újabb lehetőségeink adódnak:

1. FULL - teljes képernyő tárolása
2. TOP - felső 1/3 tárolása
3. MID - középső 1/3 tárolása
4. BOT - alsó 1/3 tárolása (kivétel 22., 23. sorok)

Pl. úgy döntöttünk, hogy a '2' billentyű megnyomásával csak a kép felső 1/3-át tároljuk el, ekkor ismét két opciót kapunk:

S: attribútumok nélküli kép (SCREEN ONLY)

A: kép az attribútumokkal együtt (+ATTRIBUTES)

Nyomjunk pl. 'A'-t, feltételezve, hogy szükségünk van az attribútumokra is, ekkor ismét két opciót kapunk, amely arra utal, hogy a tárolás kétféleképpen történhet:

N: normál tárolás, amely akkor lényeges, ha a kép gyors visszahívása elengedhetetlen (pl. animáció esetén)

C: tömörített tárolás, amely minimálisra csökkenti (sebességcsökkenés árán) a tárolási területet (COMPRESSED)

Takarékoskodjunk a memóriával, válasszuk a 'C' opciót. Ismét választás előtt állunk. Megvizsgálhatjuk a szükséges byte-ok számát (COUNT), kimenthetjük a képet tárolási célra (STORE), vagy érvényteleníthetjük kiválasztásunkat, és visszatérhetünk a főmenühöz. Mi most tekintsük meg a tárolási byte-ok számát a 'C' megnyomásával. Várható, mivel csak 1/3 képernyőt tárolunk, és azt is tömörített formában, hogy a memóriamegtakarítás igen jelentős lesz. 'S' megnyomásával a grafika eltárolódik, a vezérlés pedig visszaadódik a CATALOGUE számára. Ilyenkor tovább folytathat-

juk a grafikák eltárolását, egészen a memória határáig, ám előbb meg kell oldanunk a teljes állomány kimentését is ahhoz, hogy programjainkban később használni tudjuk. Ez igen egyszerű, ugyanis az alapmenüből ki kell választanunk a negyedik (SAVE FILE) opciót, majd követjük az egyértelmű instrukciókat. A teljes kép-állomány gépi kódként kerül kimentésre, beleértve a visszahíváshoz szükséges információkat is. Ehhez a technikához viszont be kell tartani néhány alapszabályt:

Mindenekelőtt a RAMTOP-ot állítsuk be CLEAR 29699-cel, és töltsük be az egész screen file-t. Bármely kép, vagy képrészlet megjelenítéséhez a POKE 23728,L: POKE 23729,H: RANDOMIZE USR 30000 (+ENTER) utasítás-szerkezet szükséges, és a képünk már a képernyőn is van. A szükséges L és H értékek a katalóguslistából vehetők át. Ha arra kerülne sor, hogy saját programunkból kell kimenteni a hívókédot és a SCREEN\$ állományt, akkor az az eljárás, hogy kiadjuk a SAVE "név" CODE 30300,hossz utasítást, ahol a 'hossz' értéke:

(felső harmad tárcím
+ felső harmad hossz)
- 30300

A kép állományok áthelyezése

A képeket a képernyőre visszaalakító gépi kódú rutin a 30000. memóriacimen kezdődik, a legkisebb állomány-tárolási cím pedig a 30300. Ha vigyázunk rá, hogy nem mozditjuk el, vagy nem írjuk felül a visszahívó kódot, ill. kódokat, melyek a 30000-30299 memóriaterületen helyezkednek el, mód van az egyes képek áthelyezésére is. A BASIC program hosszától függően egyes képek a 30000. cím alatt is tárolásra kerülhetnek. Ehhez feltétlenül ismernünk kell az adott állomány startcímét és hosszát, ezt a katalógusból vehetjük. Az alábbi kis program elvégzi a byte-ok mozgatását. A 'regi' jelentése a jelenlegi, az 'új' jelentése a megcímzett új cím, míg 'hossz' jelenti a file hosszát.

```
FOR n=0 TO hossz:
  POKE (uj + n),PEEK (regi + n): NEXT n
```

A további POKE utasításokhoz szükséges értékeket így számíthatjuk ki:

$$L = uj - INT (uj / 256) * 256$$

$$H = INT (uj / 256)$$

Ezt követően a RANDOMIZE USR 30000 utasítás hívja ki a képernyőre az áthelyezett állományt.

A SCREEN MACHINE ide vonatkozó hibái

Betöltéskor nem árt óvatosnak lenni. Ha tévedésből nem megfelelő képet, vagy állományt töltöttünk be, vagy a programot BREAK-kel megállítjuk, újraindításkor ne adjunk ki RUN utasítást, mert a katalógusunk törlődik. Helyette GO TO 100 a helyes parancs. RUN természetesen kiadható, de csak akkor, ha újra kívánjuk indítani az egész programot.

A képállományok áthelyezése a memória magasabb területeire

Amennyiben csak egy-két állományra van szükségünk, programunknak viszont több hely kell, akkor az állományok a következőképpen helyezhetők át:

- Mentsük ki az állományt szalagra a SCREEN MACHINE ONE segítségével normális formában, de úgy, hogy minden állomány kezdőcímét és hosszát jegyezzük fel gondosan.
- Számítsuk ki mekkora hely kell ezeknek az állományoknak a memóriában, ezt úgy tudjuk elvégezni, hogy összeadjuk az állományok hosszát, hozzáadunk 600-at és az így kapott számot levonjuk a rendelkezésre álló legmagasabb memóriacimből, ami (UDG mellőzése esetén) 65535.
- Hozzuk lejjebb a RAMTOP-ot a kód védelmére CLEAR utasítással, az itt megjelelő érték könnyen kiszámítható: új hely - 1, majd töltsük be az állományt 'LOAD "név" CODE új hely' utasítással. Mivel az áthelyezett kód első -300 byte-nyirészére nincs szükségünk, hiszen ez még a régi címeket és hosszokat tartalmazta, feljebb vihetjük a RAMTOP-ot 300 byte-tal: CLEAR új hely + 299, természetesen csak akkor, ha erre feltétlenül szükség van.
- Az áthelyezett kód most már az 'új hely + 300' címen kezdődik, az első SCREEN\$ file pedig 'új hely + 600'-on.
- Kiszámítjuk az új L és H értékeket a hívó kód számára, az alábbi összefüggés alapján:

$$L = CIM - INT (CIM / 256) * 256;$$

$$H = INT (CIM / 256)$$
 ahol 'CIM' bármelyik kép-állomány kezdőcíme.

Végül a SCREEN MACHINE ONE utolsó opciója a 'LOAD', amely lehetővé teszi, hogy korábban kimentett összetett kép-állományokat további kiegészítések céljából be tudjunk hívni a programba. Ez az alapmenü '3' opciójával lehetséges.

A Beta Basic átalakítása EPROMégetéshez

Bizonyára a Beta Basic felhasználók többségének fejében megfordult már az a gondolat, hogy milyen jó lenne ezt a színvonalas BASIC bővítőt állandósítani a számítógép mellé. Erre a legjárhatóbb út a program EPROM-ba történő égetése, majd annak adott pillanatban a memóriába történő beolvasása. Mint tudjuk az EPROM-ból történő beolvasás igen rövid időt vesz igénybe, azt is mondhatnánk az egyik pillanatról a másikra megjelenik a program a képernyőn.

A HARDWARE

A mi általunk felhasznált berendezés a 'Játék és Program Spectrum' c. könyvsorozat I. kötetének borítóján is meghirdetett DEMAK gyártmányú 16K-s külső programtár ill. epromégető volt. Az égetést a következőképpen hajthatjuk végre:

1. A max. 16K-s és kizárólag BASIC nyelvű programunkat betöltjük a gépbe, majd utánatöltjük az égetővel együtt forgalmazott kazettáról a 'SETUP' nevű rutint. Ez a rutin - amennyiben meghívjuk - új RAMTOP-ot állít be, és programunkat a RAMTOP felé helyezi, előkészítve a beégetéshez. Ekkor a RAMTOP rendszerváltozó tartalmát kiolvassva a RAMTOP feletti részt kódként ki kell mentenünk kazettás magnetofonra.
2. Csatlakoztatjuk a számítógéphez az epromégetőt, majd betöltjük az égető programot, ezt követően pedig a 'SETUP' segítségével előkészített programot is. Behelyezzük az epromot, és elvégezzük az égetést.

Az így elkészített epromot a külső programtárba helyezzük, majd a programtárat a géphez illesztjük, és bekapcsoláskor már a programunk jelentkezik be a képernyőn.

A SOFTWARE

Mivel a 16K-s DEMAK rendszer csak BASIC programokat tud kezelni, a Beta Basic programot megfelelően át kell alakítanunk. A Beta 1.8 0. sorából és a mintegy 9.5K gépi kódú részből egy teljesen szabályos BASIC programot kell előállítanunk, minden, BASIC által nem értelmezhető részt REM mögé helyezve. A második lépésben - mivel a külső programtár semmi mást nem tesz, csak az így előkészített programot az epromból a BASIC te-

rületre tölti -, gondoskodni kell az eredeti állapot visszaállításáról is.

A legnagyobb problémát a gépi kód REM sorának elkészítése jelenti. Első megközelítésre a dolog reménytelennek tűnik, mivel a gép egy képernyő teleírása után csak figyelmeztető bűgás után veszi be a következő karaktert. Igaz ugyan, hogy az így beírt karakter sem vész el, sőt a SPECTRUM - mivel a sor hosszát két byte-on tárolja - képes igen hosszú sorok generálására is, az nagyon hosszadalmas lenne, ha 9734 karakter hosszúságú REM sort szeretnénk generálni.

Talán hihetetlennek tűnik, de a problémát maga a program, a Beta Basic fogja megoldani.

```

99 REM * az LDIR helye *
100 LET a$=STRING$(15,"r")
110 LET b$="5 REM "+a$
120 KEYIN b$
130 LET a$=STRING$(5000,"1")
140 LET b$="10 REM "+a$
150 KEYIN b$
159 REM * Beta 2.rész *
160 LET a$=STRING$(4732,"2")
170 LET b$="20 REM "+a$
180 KEYIN b$
190 JOIN 10: REM * 10+20 sorok egybe *
199 REM * Beta a REM sorba *
200 POKE 24304,MEMORY$(55801 TO)
209 REM * LDIR beírás *
210 FOR n=24283 TO 24294
220 PRINT TAB 0;n: INPUT "poke: ";m
230 POKE n,m: PRINT TAB 15; PEEK n
240 NEXT n
249 REM * 0. sor átalakítása *
250 POKE 23756,1: POKE 23759,234

```

A 100. sorban előállítunk a STRING\$ függvény segítségével egy 15 karakter hosszú 'a\$' szöveges változót. A 110. sorban ezt hozzáadjuk az "5 REM " b\$ változóhoz, és beolvastatjuk a géppel a 120. sorban a 'KEYIN' utasítás segítségével (a REM-et kulcsszóként kell beírni!). Ha ezt futtatjuk, az eredmény 'C Nonsense in BASIC' hibaüzenet lesz, valamint az 5. sor 15 db. 'r' betűvel (A hibaüzenet okáról egyébként cikket olvashattunk a BITLET 40. számában). A REM sor egy későbbi LDIR rutinhoz kell. Láthatjuk, hogy így is lehet REM sort készíteni, valamivel gyorsabban, mint az előző módszerrel.

Előző sikerünkön felbuzdulva a 130-150. sorokban állítunk elő egy 10-es REM sort, amely 5000 (!) db. 1-est tartalmaz, majd a

160-180. sorokban egy 20. REM sort 4732 db. 2-essel. A 200-as sorban a 'JOIN' utasítás segítségével a két REM sort összetoljuk, ezzel lesz egy 10-es sorunk, ami 5000+4732+REM+kettőspont=9735 karakter hosszú. Ezt a programsort azért kellett két lépésben elkészíteni, mert egy 9735 karakter hosszúságú 'a\$' nem fér el a változóterületen.

Most futtassuk le a programot a 190. sorig (RUN 100: RUN 130: RUN 160) a hibaüzenetek miatt. Ezzel elkészítettük a programunkhoz szükséges REM sorokat.

FIGYELEM! Az ezután következő memóriacímek BASIC sorok címei. Beírásuknál nem lehet a gépen Interface I, és a 0. Beta Basic sornak a memóriában kell lennie.

A 200-as sorban egyetlen egy kiterjesztett POKE utasítással elhelyezzük a Beta Basic gépi kódját a 10. sorba, közvetlenül a REM utasítás utáni karakter címére.

A 210-240. sorokkal beírhatjuk decimálisan azt az LDIR rutint, amely futtatásakor helyre fogja tenni a Beta Basic gépi kódját. Helye az 5-ös REM sor.

LD DE,55801	17	249	217
LD HL,24304	33	240	94
LD BC,9734	1	6	38
LDIR	237	176	
RET		201	

A 250-es sor átalakítja a Beta Basic függvényeit tartalmazó 0. sort. A sorszámot 1-re javítja (POKE 23756,1), és az első DEF FN utasítást REM-re változtatja (POKE 23759,234).

0 DEF FN R(N)=USR 0:.....
.....REM: RANDOMIZE USR 0

1 REM R(N)=USR 0:.....
.....REM: RANDOMIZE USR 0

A REM sorok feltöltését is elvégeztük, kiadhatjuk a 'DELETE 99 TO' parancsot, erre a programrészre már nem lesz többet szükségünk.

A most következő programrész már beégetésre fog kerülni. Feladata a következő lesz: Autostarttal indul, meghívja az LDIR rutint, ezzel helyre teszi a gépi kódot. Meghívja a Beta Basic-et, kitakarítja az egész BASIC területet, kivéve a 0. sort, amit előbb még helyre kell állítania. Kikötés még, hogy nem tartalmazhat Beta Basic utasítást. Írjuk be a REM sorok alá a következő listát.

```
100 GO TO 200
110 CLEAR 55800
112 LET c=PEEK 23635+256*PEEK 23636
114 LET b=c+549: POKE c+532,b-256*INT
(b/256): POKE c+533,INT (b/256)
120 RANDOMIZE USR (c+528)
130 POKE c+1,0: POKE c+4,206
140 POKE 23624,63: POKE 65533,68: POKE
65534,56
150 RANDOMIZE USR 58419
160 POKE 23624,56: POKE 23609,10: NEW
200 CLS: PRINT #0;"© 1988 Beta Basic 1.8E
SpVsoft": PAUSE 0
210 IF INKEY$ <> "y" THEN GO TO 110
```

A program bekapcsolás, vagy a RESET gomb megnyomása esetén autostarttal fog indulni, miután a külső programtár a memóriába töltötte. A REM sorokat átugorva, a 100-as sorról a 200-asra ugrik, ahol saját copyright felirattal bejelentkezik a képernyő alsó részén, és billentyűlenyomásra vár. Ha itt 'y' billentyűt nyomunk (210. sor), semmi sem történik, a program '0 OK, 300:1' üzenettel meg fog állni.

Más billentyűt lenyomva a 110-es sorhoz jutunk. Itt beállítjuk a RAMTOP-ot. A 112-es sor minden eshetőségre számítva megkeresi a BASIC terület kezdőcímét (pl. Interface I). A 114-es sor az így kiszámított 10-es REM sor címét írja az LDIR rutinba, amit a 120-as sor meg is hív. Ezzel a gépi kód a helyére került.

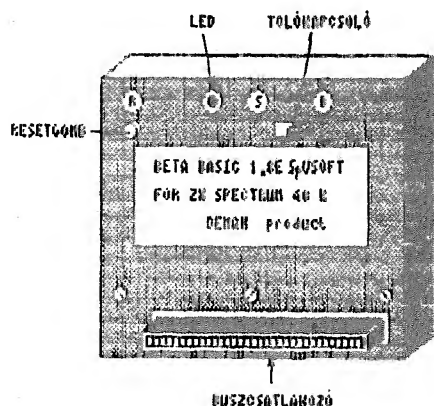
A 130-as sor helyreállítja a 0. sort. A 140. sor első utasítása lefordítva: BORDER 7: BORDER INK 7. Hatása: Igaz, hogy a Beta Basic bejelentkezik a kereten, de fehér tinta színnel, ezért nem lesz látható. A 2. és 3. utasítás: az UDG területen elhelyeztük a magyar ABC betűit. A MEMORY\$ függvény a memória első és utolsó 2 byte-ját nem kezeli, ezért ezzel a két POKE utasítással az 'ú' betűt javítottuk ki. A 150-es sortól már Beta Basic-ben vagyunk. A 160. sorban a keretet visszaállítjuk normális értékre, a billentyűhangot is beállítjuk, majd kiadjuk a NEW utasítást. Ez már a Beta Basic-ben kiadott NEW, a 0. sor ép marad.

Az így elkészített program már működőképes. Futtatás előtt azért nem árt magnetofonra menteni (SAVE "eprbeta" LINE 100).

A RUN parancs kiadása után a program copyright felirattal bejelentkezik, s egy billentyű megnyomása után megjelenik a képernyő alján a villogó 'K' kurzor is. Ilyenkor, ha ENTER-t nyomunk, elő kell tűnnie a 0. sornak is. Ha a program megfelelően működik, elővehetjük a 'SETUP' rutint, és hozzákezdhetünk az égetéshez.

Az EPROM a gyakorlatban

A programtárat kikapcsolt állapotban a gép hátsó 'él-csatlakozójára' kell illeszteni. Bekapcsolás után a LED pillanatnyi felvillanása jelzi, hogy a programtár az EPROM tartalmát a Spectrum memóriájába töltötte.



A külső tár kezelése rugalmas, ha a teljes memóriára szükségünk van, akkor a programtár bármikor levehető. Ha a tárban 16K-s EPROM van, a toló-kapcsoló bal oldali helyzetében a RESET megnyomása után a normál BASIC jelentkezik be, míg jobb oldali állásban a Beta Basic.

Ha a Beta Basic bejelentkezett, 'y' megnyomásakor még mindig a normál BASIC-ben maradhatunk, ekkor egy NEW kiadása után ismét használható a gép. Ha már Beta Basic-ben vagyunk, akkor is bármikor kiadhatjuk a 'RANDOMIZE USR 0' parancsot, ez a Beta Basic-et is kikapcsolja. Ezután CLEAR USR "a"-1, és ismét NEW. Az eredmény pedig az alap BASIC lesz, de a magyar ékezetes betűkészlettel.

A 4. lista 112. sorának köszönhetően a program akkor is működőképes, ha a BASIC terület kezdőcíme, és ezzel együtt a REM sorok címe is valami miatt eltolódott (Interface I, Microdrive kompatibilis, ez már a gyakorlatban is beigazolódott).

OKFEJTÉS

Miért pont az 1.8 változat?

Először is ezzel a programtárral max. 16K-s programok kezelhetők, így a 3.0 már nem fért volna bele. A 64K-s programtár már lényegesen több költséget jelent, és hosszú ideig nem is volt működőképes. Másodszor nekünk ez a verzió a legszimpatikusabb, a szabad memória 31351 byte, így bőven marad hely programírásra. A 3.0 igaz, hogy sok többlétszolgáltatást nyújt, de azon kívül, hogy több helyet foglal, egy kicsit el is lett bonyolítva, és több a hibája is. Néha pl. ha valamelyik ablak aktiv, és programot töltünk be magnetofonról, előfordulhat, hogy a fejléc beolvasása és a programnév kiírása után a gép 'scroll?' üzenettel leáll. Az is könnyen előfordulhat, hogy minden látható ok nélkül elszáll a Beta 3.0, magával repítve több órai munkánkat is. A 4.0 verzió egyébként egyelőre nincs még a birtokunkban.

Miért ebben a formában égettük be?

Azért nem definiáltunk pl. billentyűket, mert egy CLEAR utasítás után (amit elég gyakran használ az ember), minden definiált billentyű tartalma elvész, lehet kezdeni az egészet előről. A mi általunk égetett forma egyébként az alap 1.8 változat, kiegészítve a magyar UDG karakterekkel, amelyek KEYWORDS 0 üzemmódban elérhetők. Indításkor a rendszer KEYWORDS 1 üzemmódban jelentkezik be, AUTO parancsra a sorszámozás 10-től indul, és tízesével növekszik.

Athena

Az örökélet bevételéhez a BASIC/6912/40665/... file-térképű verziót választottuk.

Töltsük be a loader-t, majd állítsuk le a magnetofont. RESET, és írjuk be a következő BASIC sorokat:

```
10 FOR i=23296 TO 23309: READ a: POKE i,a: NEXT i
20 DATA 55,62,255,221,33,217,158,17,38,97,205,86,5,201
30 CLEAR VAL "24399": LOAD ""CODE 16384: RANDOMIZE USR 23296
40 POKE 48853,0: POKE 48861,0: POKE 51661,0: POKE 52593,0: POKE 55594,0: RANDOMIZE USR 61632
```

Futtassuk a programot (RUN), és indítsuk tovább a magnetofont.

Chronos

A végtelen lövés bevitele nagyon egyszerű. Az "Input name" kérdésre írjuk be a következőket:

JING IT BABY

Ekkor a nevünk nem kerül be a ponttáblázatba, viszont a játék visszatér a menübe, és megjelenik egy új opció. A '7'-es billentyűvel a szuper lezert kapcsolhatjuk be/ki.

Az előző alkalommal megismerhettük, hogy ez az önmagában nagyszerű periféria milyen menü-funkciók kiváltásával segíti elő a hatékony nyomtatási lehetőséget. Most azt tekintjük át, hogyan vezérelhető BASIC-ből, milyen módon oldható meg segítségével a lapozás, és hogyan illeszthető össze az egyes ismert rendszerprogramokkal.

A MULTIPRINT vezérlése BASIC utasításokkal:

A standard BASIC nyomtatási parancsok 95%-ban üzemképesek, LPRINT - minden helyzetből probléma nélkül működik.

LLIST - hosszabb programok listázása esetén előfordulhat, hogy az első sor nyomtatása hibás lesz, ilyenkor a MULTIPRINT hibásan inicializálódott, bár utólag elegendő csak az első sort újra kinyomtatni.

COPY - szintén hosszabb programok esetén okozhat gondot alkalmazása, mivel meghatározott méretű munkaterületre van szüksége a gép saját RAM területén belül. Ilyen esetben 'Out of memory...' hibaüzenetet kapunk.

Bekapcsoláskor a MULTIPRINT a következő beállított (DEFAULT) értékeket tartalmazza:

LINE FEED - ON	TOKENS - ON
TEXT COPY	WIDTH - 80 character
TOP LINE - 0	BOTTOM LINE - 23

A MULTIPRINT rendelkezik azzal az előnyös tulajdonsággal is, hogy kibővíti a gép alaputasításkészletét. A MULTIPRINT parancsait REM alatt kell elhelyezni. Az általános szintaktika a következő:

REM MP, ...

A rendelkezésre álló lehetőségeink a következők (a bemutatást tetszőleges adatok szolgálják):

REM MP, LFO - LINE FEED OFF módban
 REM MP, LFI - LINE FEED ON módban
 REM MP, WD80 - karakter-oszlopok száma: 80
 (megadható 0-255)
 REM MP, LM12 - balmargó a 12. oszlopra
 (megadható 0-255)
 REM MP, TM8 - Felső sor a lap tetejétől 8 sorral lejjebb
 (megadható 0-23)
 REM MP, BE10 - alsó sor a lap aljától 10 sorral feljebb
 (megadható 0-23)
 REM MP, CPT - TEXT COPY beállítása
 (csak ASCII karakterek)

REM MP, CPN - NORMAL COPY beállítása
 (7x11 cm méretű HI-RES copy)

REM MP, CPL - LARGE COPY beállítása
 (12,7x16 cm méretű HI-RES copy)

REM MP, CPS - SHADED COPY beállítása
 (12,7x16 cm méretű shaded copy)

REM MP, LS12 - Beállítja a sorok közötti szünetet 12/72 értékre (1/6")

(megadható 0-255)

REM MP, TKO - Kikapcsol a TOKEN üzemmódból.

Ilyenkor a MULTIPRINT 0-255 értékű CHR\$ adatokat képes kiküldeni a nyomtatóra, ez lehet ESCAPE szekvencia, vagy bináris adat is. Megjegyezzük, hogy a LINE FEED automatikusan beállítódik REM MP, LFI kiadását követően. Ebben az üzemmódban az LLIST parancs nem használható.

REM MP, TK1 - Bekapcsolja a TOKEN üzemmódot

Ezt az üzemmódot kell beállítanunk, ha BASIC programot akarunk kinyomtatni az LLIST parancs segítségével. Csak az ASCII karakterek (CHR\$ 32-127) és a CARRIAGE RETURN (kocsivissza/soremelés) kód (CHR\$ 13) működik zökkenőmentesen, az egyéb kódok megjelenésekor a nyomtatási sebesség csökken, ugyanis ilyenkor a megfelelő funkció előbb dekódolásra kerül. Az ilyen speciális kódok a következők:

CHR\$ 6 - Tabuláció

Beállítja a nyomtatás kezdő pozícióját középre a bal margó aktuális értékéhez képest. Megfelel az LPRINT, parancsoknak.

CHR\$ 22 - AT vezérlő

Két numerikus paraméter követi. Megfelel az LPRINT AT Y,X parancsoknak

CHR\$ 23 - TAB vezérlő

Egy numerikus paraméter követi. Megfelel az LPRINT TAB X parancsoknak.

CHR\$ 27 - Direkt vezérlő

A CHR\$ utasítást követő kód közvetlen vezérlést hajt végre a nyomtató számára. Pl. nyomtató RESET az ESCAPE "v" parancsal lehetséges, ez egyébként a CHR\$ 27+"v" láncsal írható le. A MULTIPRINT felhasználásakor ez a következőképpen néz ki: CHR\$ 27; CHR\$ 27; CHR\$ 27; "v"

CHR\$ 128-143 - Blokk gr. karakterek

Ha TEXT COPY-t kérünk, ezeknek a karaktereknek a helyén SPACE jelenik meg. Máskülönb a grafikus karakter megjelenik, de ennek alapvető feltétele egy pontmátrix nyomtató felhasználása.

CHR\$ 165-255 - BASIC kulcsszavak

Nyomtatáskor a megfelelő BASIC utasítás kulcsszava is reálisan jelenik meg a megfelelő kód alapján.

REM MP, PDn, 27, 82, 3 - Közvetlen nyomtatóvezérlés definiálása kódlánc segítségével.

Az itt látható példa az ESC 82.3 kód, amely EPSON nyomtatókon kiválasztja az angol karakterkészletet.

REM MP, PDn, 0 - Figyelmen kívül hagyja az 'n' definiált vezérlő kódot.

REM MP,PDn,1 - Aktív állapotba helyezi az 'n' definiált vezérlő kódot.

Megjegyzés: 'n' értéke 0-5 között lehet. A definíció után elegendő az adott vezérlést LPRINT CHR\$ n utasítással kiváltani, ahol 'n' az adott definícióhoz rendelt érték.

10 REM MP,PD1,27,52

Definiálunk egy '1' kódláncot, amely megfelel az ESC "4" szerkezetnek, azaz az EPSON kompatibilis nyomtatókon beállítja az ITALICS készletet.

20 REM MP,PD2,27,53

Definiálunk egy '2' kódláncot, amely megfelel az ESC "5" szerkezetnek, azaz az EPSON kompatibilis nyomtatókon kikapcsolja az ITALICS készletet.

```
30 FOR n=1 TO 2: GO SUB 50: NEXT n: STOP
50 LPRINT CHR$ n; "SPECTRUM VILAG"
60 RETURN
```

A program eredményeképpen a 'SPECTRUM VILAG' felirat egyszer normál, egyszer pedig ITALICS betűkkel nyomtatódik ki a nyomtatón.

Az egyes parancsok egy REM alatt is elhelyezhetők, ilyenkor vesszővel kell ezeket elválasztanunk:

```
REM MP,LF1,WD20,TK1
```

Amennyiben érvénytelen tartományba esik a megadott paraméter, a gép 'Q Parameter error' hibaüzenettel válaszol.

A MULTIPRINT összes változója (beleértve a PD vezérlő-láncokat is), a MULTIPRINT saját 8K RAM-jában tárolódik, így ezek egy esetleges NEW, vagy rendszer RESET következtében sem sérülnek meg. Az alaphelyzetben beállított (DEFAULT) értékek mindig visszaállíthatók egy REM MP (+ENTER) utasítás kiadását követően.

A MULTIPRINT PROGRAMOZÁSA

Azokban a verziókban, amelyekbe már beépítették a (J)ump opciót, lehetőségünk van közvetlen gépi kódú belépéseket végeznünk, ez elvégezhető a MULTIFACE PROGRAMOZÁSA szerint, amelyről a 'SpV' 7. részében esett szó. A MULTIPRINT is az első memórialapot lapozza felül (0-16384/0000h-4000h). A lapozás a MULTIFACE összes verziójával ellentétben új portcímeken történik:

```
IN A,(187) ill. IN A,(BBh) - belapozás
IN A,(191) ill. IN A,(BFh) - kilapozás
OUT A,(187) ill. OUT A,(BBh) - NMI biztosítása
OUT A,(191) ill. OUT A,(BFh) - NMI biztosítása
```

A MULTIPRINT software kompatibilitása

Mielőtt bármilyen software-t betöltenénk, inicializáljuk a nyomtatót: LPRINT (ENTER).

TASWORD 2, 3 és +2

Mielőtt betölthük a TASWORD-öt, adjuk ki: LPRINT: REM MP, TK0 (+ENTER).

Célszerű ezt az inicializálást beépíteni a BASIC LOADER-be. TASWORD 2 esetén a 'G' opciót követően (ld. 'JÁTÉK ÉS PROGRAM SPECTRUM IV. 109. old. - LSI ATSz, 1987) adjuk meg a következő kódot:

```
CODE 1 : 0
CODE 2 : 0
CODE 3 : 0
CODE 4 : 57855
```

OMNICALC 2

Betöltése előtt állítsuk át a MULTIPRINT megfelelő opcióját NORMAL-, vagy LARGE COPY állapotba, vagy ugyanezt írjuk be a LOADER-be, megfelelő REM MP,... segítségével.

OCP software-ek

Válasszuk ki a 'KEMPSTON E' opciót.

VTX 5000 MODEM + software

Nyomjunk BREAK-et a VTX főmenüjében, majd REM MP,... segítségével adjuk meg a HI-RES COPY valamely változatát.

GENIE DISASSEMBLER

A GENIE BASIC LOADER-ébe írjuk be a következő sort:

```
1 POKE 28815,211: POKE 28816,191: LPRINT:
REM MP,TK0
```

Amikor a GENIE megkérdezi a Page In/Out adatokat, In-re írjuk be a 187-et, Out-ra pedig a 191-es értéket.

Megjegyzések: A MULTIPRINT varázsgombját soha ne nyomjuk meg, ha az INTERFACE I.-en keresztül kommunikációt végzünk (pl. MICRODRIVE-ra).

A COPY paraméterek a MULTIPRINT RAM 8240 (2030h) címétől tárolódnak el pl. 27,75,0,1 - NORMAL COPY, 27,76,0,3 - LARGE COPY.

Enduro Racer

Töltsük be a programot, és indítsuk el a játékot. Ezután nyomjuk meg a 'Q' és a 'CAPS SHIFT' billentyűket egyszerre. Ekkor mindenben keresztül mehetünk teljes sebességgel, csak kormányozni kell. Ez a kis trükk csak a harmadik pályáig jó, a következő pályákon már nekünk kell boldogulni. A szupersebességhez megpróbálhatjuk a W-Q-A-Z billentyűk együttes lenyomását is.

A hanggenerátor programozása

Ahhoz, hogy megszólaltassuk a hanggenerátort, először is ki kell nyitni a kimeneti csatornákat. Ha tiszta hangot akarunk, ki kell kapcsolni a zajt ($R7 = 56$). Válasszunk mind a három csatornára közepes hangerő-értéket ($R8-R10 = 8$), majd adjunk tetszőleges értéket a hangmagasságoknak ($R0, R2, R4$). A minta-program a következőképpen néz ki:

```
10 OUT 65533,7: OUT 49149,56
20 OUT 65533,8: OUT 49149,8
30 OUT 65533,9: OUT 49149,8
40 OUT 65533,10: OUT 49149,8
50 OUT 65533,0: OUT 49149,200
60 OUT 65533,2: OUT 49149,133
70 OUT 65533,4: OUT 49149,103
```

Futtatás után azt tapasztaljuk, hogy a három hang megszólal, de elég hamisan.

Szólaltassunk meg egy szép dó-mi-szó (C-E-G) hangzást a 4. oktávban. Előbbi programunkat módosítsuk az 50. sortól kezdve:

```
50 OUT 65533,0: OUT 49149,144
55 OUT 65533,1: OUT 49149,2
60 OUT 65533,2: OUT 49149,8
65 OUT 65533,3: OUT 49149,2
70 OUT 65533,4: OUT 49149,182
75 OUT 65533,4: OUT 49149,1
```

Futtassuk ismét a programot és azt, tapasztaljuk, hogy a hangok **konzonánsak**, egymáshoz illenek.

Nullázzuk ki a hanggenerátor regisztereit, majd NEW paranccsal töröljük a programot is. A következő mintaprogram a csilingelő harangok vagy az üvegpoharak hangjára emlékeztető hangot szólaltat meg:

```
10 OUT 65533,0: OUT 49149,200
20 OUT 65533,1: OUT 49149,1
30 OUT 65533,2: OUT 49149,130
40 OUT 65533,4: OUT 49149,60
50 OUT 65533,7: OUT 49149,56
60 OUT 65533,8: OUT 49149,16
70 OUT 65533,9: OUT 49149,16
80 OUT 65533,10: OUT 49149,16
90 OUT 65533,12: OUT 49149,50
100 OUT 65533,13: OUT 49149,8
```

Most módosítsuk a következő sorokat, hogy egy pisztolylövéshez hasonlító hang szólaljon meg:

```
50 OUT 65533,7: OUT 49149,7
90 OUT 65533,12: OUT 49149,7
100 OUT 65533,13: OUT 49149,9
110 PAUSE 0: GO TO 10
```

A 110. sor jelentősége abban áll, hogy ha folyamatosan lenyomva tartunk egy billentyűt, egy gépfegyver zaját szimulálhatjuk.

Újabb módosításokkal a programunk egy közeledő gőzmozdony hangját produkálja:

```
1 LET A=31
10 OUT 65533,0: OUT 49149,200
20 OUT 65533,1: OUT 49149,7
30 OUT 65533,2: OUT 49149,130
35 OUT 65533,3: OUT 49149,7
40 OUT 65533,4: OUT 49149,60
50 OUT 65533,7: OUT 49149,31
60 OUT 65533,8: OUT 49149,16
70 OUT 65533,9: OUT 49149,16
80 OUT 65533,10: OUT 49149,16
85 OUT 65533,6: OUT 49149,INT A
90 OUT 65533,12: OUT 49149,12
100 OUT 65533,13: OUT 49149,9
105 LET A=A-0.1: IF A<1 THEN STOP
110 FOR I=1 TO 10: NEXT I: GO TO 10
```

Két csatornán szólaltassunk meg egymástól eltérő magasságú hangokat, a harmadikon pedig egy dob gép alapot. Ehhez a következő program futtatása szükséges:

```
10 OUT 65533,0: OUT 49149,36: OUT
65533,1: OUT 49149,1
20 OUT 65533,2: OUT 49149,32: OUT
65533,3: OUT 49149,9
30 OUT 65533,4: OUT 49149,144: OUT
65533,5: OUT 49149,4
40 OUT 65533,7: OUT 49149,56
50 OUT 65533,8: OUT 49149,12
60 OUT 65533,9: OUT 49149,11
70 OUT 65533,10: OUT 49149,16
80 OUT 65533,12: OUT 49149,5
90 OUT 65533,13: OUT 49149,8
```

Az alábbi program a véletlenszám-generátor felhasználásával szemléltet érdekes hatást:

```
10 OUT 65533,7: OUT 49149,56
20 OUT 65533,8: OUT 49149,INT (RND*15)
30 OUT 65533,9: OUT 49149,INT (RND*15)
40 OUT 65533,10: OUT 49149,INT (RND*15)
50 LET A=INT (RND*255): LET B=INT
(RND*15)
```



```

60 LET C=INT (RND*255): LET D=INT
  (RND*15)
70 LET E=INT (RND*255): LET F=INT
  (RND*15)
80 OUT 65533,0: OUT 49149,A: OUT
  65533,1: OUT 49149,B
90 OUT 65533,2: OUT 49149,C: OUT
  65533,3: OUT 49149,D
100 OUT 65533,4: OUT 49149,E: OUT
  65533,5: OUT 49149,F
110 GO TO 20

```

Mindezek után elérkezett az ideje, hogy valamilyen felismerhető dallamot is eljuttassunk. Használjuk ki a három csatornát, s egy ismert dallamot játsszunk el kánonban, vagyis az azonos – de különböző oktávon – megszólaló dallamok egymás után folyamatosan lépnek be, így igen érdekes hatást érünk el.

A vezérlő szakasz a 10–50 sorok között lett elhelyezve, a 100. sortól találjuk az adatokat.

```

10 READ a
20 IF a=16 THEN OUT 65533,7: OUT
  49149,63: STOP
30 IF a>16 AND a<21 THEN LET a=a-16:
  PAUSE 40/a: GO TO 10
40 READ b
50 OUT 65533,a: OUT 49149,b: GO TO 10
100 DATA 7,56,8,15,9,15,10,15
110 DATA 0,235,1,1,18,0,182,1,1,18,0,134,1,1,
  18,0,235,1,1,18
115 DATA 0,235,1,1,18,0,182,1,1,18,0,134,1,
  1,18,0,235,1,1,18
120 DATA 0,134,1,1,2,214,3,3,18,0,112,1,1,2,
  107,3,3,18,0,72,1,1,2,12,3,3,18,2,214,3,3,18
125 DATA 0,134,1,1,2,214,3,3,18,0,112,1,1,2,
  107,3,3,18,0,72,1,1,2,12,3,3,18,2,214,3,3,18

```

```

130 DATA 0,72,1,1,2,12,3,3,4,246,5,0,20,0,36,
  1,1,20,0,72,1,1,2,224,3,2,4,219,5,0,20,0,112,
  1,1,20,0,134,1,1,2,144,3,2,4,195,5,0,18,0,
  235,1,1,4,246,5,0,18
135 DATA 0,72,1,1,2,12,3,3,4,246,5,0,20,0,36,
  1,1,20,0,72,1,1,2,224,3,2,4,219,5,0,20,0,112,
  1,1,20,0,134,1,1,2,144,3,2,4,195,5,0,18,0,
  235,1,1,4,246,5,0,18
140 DATA 0,235,1,1,2,144,3,2,4,195,5,0,20,2,
  72,3,2,20,0,144,1,2,2,144,3,2,
  4,184,5,0,20,2,224,3,2,20,0,235,1,1,2,12,
  3,3,4,164,5,0,18,2,214,3,3,18
145 DATA 0,235,1,1,2,144,3,2,4,195,5,0,20,2,
  72,3,2,20,0,144,1,2,2,144,3,2,
  4,184,5,0,20,2,224,3,2,20,0,235,1,1,2,12,3,
  3,4,164,5,0,18,2,214,3,3,18
150 DATA 0,0,1,0,2,214,3,3,4,164,5,0,20,4,146,
  5,0,20,2,31,3,5,4,164,5,0,20,
  4,184,5,0,20,2,214,3,3,4,195,5,0,18,4,246,
  5,0,18
155 DATA 0,0,1,0,2,214,3,3,4,164,5,0,20,4,146,
  5,0,20,2,31,3,5,4,164,5,0,20,
  4,184,5,0,20,2,214,3,3,4,195,5,0,18,4,
  246,5,0,18
160 DATA 2,0,3,0,4,246,5,0,18,4,72,5,1,18,4,
  246,5,0,17
165 DATA 2,0,3,0,4,246,5,0,18,4,72,5,1,18,4,
  246,5,0,17
170 DATA 4,0,5,0,16

```

Az adatokat könnyen elgépelhetjük, így végezzük el a következő ellenőrzést:

```

LET z=0: FOR i=0 TO 470: READ a: LET z=
z+a: NEXT i: PRINT z

```

Ha az eredményünk 17240, úgy minden rendben van, máskülönben nem árt ellenőrizni a listát!

Gunrunner

Az örökélet és a sérthetatlenség beviteléhez a BASIC/768/6912/40535 file-térképpel rendelkező verziót választottuk ki.

Töltsük be a loader-t, majd RESET, és írjuk be a következőket:

```

10 CLEAR 25317: LOAD ""CODE: POKE 64531,68: RANDOMIZE USR 64512
20 POKE 64520,12: FOR f=23308 TO 23323
30 READ a: POKE f,a: NEXT f
40 DATA 175,50,19,192,50,132,190,50,169,192,50,69,205,195,198,187
50 RANDOMIZE USR 65082

```

Futtassuk a programot (RUN), és indítsuk el a magnetofont.

Sidewalk

A végtelen energia bevitelét a BASIC/6912/37120 file-térképpel rendelkező verzióra közöljük.

Töltsük be a loader-t és RESET-eljük a gépet. Írjuk be a következő programot:

```

10 CLEAR 24831: LOAD ""SCREEN$: LOAD ""CODE: POKE 60234,0: RANDOMIZE USR 51200

```

Indítsuk el a programot (RUN), és ezután a magnetofont is.

VII. UGRÓ UTASÍTÁSOK (FOLYT.)

Az elmúlt alkalommal megismertük a feltétel nélküli ugró utasításokat, most azokat tekintjük át, amelyek csak valamely előzetes feltétel teljesülése esetén hajtódnak végre. A feltételhez kötött ugró utasítások minden esetben valamely jelzőbit állapotát vizsgálják. Az egyes alcsoportokat is eszerint különböztetjük meg.

4. Az átviteli jelzőbit állapotától függő ugrás

Négy olyan utasítást különböztetünk meg, amely csak akkor hajtja végre az ugrást, ha az átviteli jelzőbit állapota az utasítás szerkezetének megfelelő.

Az átviteli jelzőbitről a kezdet kezdetén már volt szó. Pár szóban azért ismételjük át a vele kapcsolatos tudnivalókat. Az átviteli jelzőbit az 'F' regiszter 0. bitje (CARRY FLAG = C). Ez jelzi a bináris túlcscordulást. Az összeadó és kivonó utasításoknál már láttuk, hogy több utasítás (pl. ADD, ADC, SUB, SBC) túl-, ill. alulcscordulás esetén 1-be állítja ezt a jelzőbitet. Azt is megismertük, hogy a logikai utasítások (AND, OR, XOR) minden esetben törlik az átviteli jelzőbitet. Néhány bitléptető utasítás (ezekről csak később lesz szó) is hatással van az átviteli jelzőbitre. Az átviteli jelzőbit állapota tehát sok esetben meghatározó lehet a program végrehajtása szempontjából. Ezt a lehetőséget használhatjuk ki elágazások végrehajtására, ugyanis a feltételes ugró utasítások valójában a BASIC IF...THEN szerkezetéhez állnak a legközelebb.

D2	X	Y	210,x,y	JP	NC,CIM
DA	X	Y	218,x,y	JP	C,CIM
	30	N	48,n	JR	NC,CIM
	38	N	56,n	JR	C,CIM

ahol x,y (Xh, Yh) a CIM alsó/felső byte-ja, n (Nh) pedig a relatív eltolási kód.

'NC' jelenti azt az állapotot, amikor az ugrás csak akkor hajtódik végre, ha az átviteli jelzőbit zérus értékű, míg 'C' esetén akkor, ha 1-es. JP és JR az abszolút és relatív ugrást jelenti, mint ahogy már azt az elmúlt alkalommal megismertük.

p1d.	40000	3E	46	62,70	LD	A,70	
	40002	FE	50	254,80	CP	80	
	40004	DA	00	80	218,0,128	JP	C,32768
	40007		C9	201	RET		

Mi is történik itt valójában? Az 'A' regiszter értékét feltöltjük 70-nel, majd összehasonlítjuk az 'A' regiszter tartalmát 80-nal. Ha visszaemlékezünk az összehasonlító utasításoknál elmondottakra, eszünkbe jut, hogy ha összehasonlításkor az 'A' regiszter tartalma kisebb, mint a konstans adatbyte, akkor a zérus jelzőbit 0 marad, de az átviteli jelzőbit 1-be vált. Az összehasonlítást követően tehát a 'C' jelzőbit 1 értékű lesz, s az ugrás megtörténik a 32768-as címre. Mielőtt futtatjuk ezt a kis kódot, ne felejtünk a 32768. címre 201-et elhelyezni, máskülönben ismét ráfut a vezérlés a 40000. címre, és a végtelen ciklusból nem tudunk kilépni (tömondatlanban lefagy a gép).

5. A zérus jelzőbit állapotától függő ugrás

Ebben a csoportban is 4 utasítást különböztetünk meg. A zérus jelzőbitről is esett már szó (ZERO FLAG = Z). Ez az 'F' regiszter 6. bitje. Ez a jelzőbit 1 értékre vált abban az esetben, ha egy művelet eredménye zérus lett, máskülönben törlik. A zérus jelzőbitre sok utasítás hatással van. Az összes olyan utasítás, amely 8 bites műveleteket hajt végre (pl. ADD, INC, SBC, CP, XOR, stb.) minden esetben 1-be állítja ezt a jelzőbitet, ha a művelet eredménye zérus lesz. A zérus jelzőbit állapotát az LD utasítások nem befolyásolják (kivéve LD A,I, LD A,R). Több olyan utasítás is hatással van a zérus jelzőbit állapotára (pl. bitléptető, blokk-kereső utasítások), amelyekkel még nem találkoztunk. Ezekről később lesz szó.

C2	X	Y	194,x,y	JP	NZ,CIM
CA	X	Y	202,x,y	JP	Z,CIM
	20	N	32,n	JR	NZ,CIM
	28	N	40,n	JR	Z,CIM

ahol x,y (Xh, Yh) a 'CIM' alsó/felső byte-ja, n(Nh) pedig a relatív eltolási kód. 'NZ' és 'Z' értelmezése az előzőekben leírtak szerint történik.

pld.	40000	0E 0F	14,15	LD C,15
	40002	0D	13	DEC C
	40003	AF	175	XOR A
	40004	B1	177	OR C
	40005	20 FB	32,251	JR NZ,40002
	40007	C9	201	RET

Ebben a mintapéldában azt szemléltetjük, hogy egészen addig visszaugrás történik a 40002. címre, amíg a 'C' regiszter értéke el nem éri a zérust. Amikor ez megtörténik, a program futása a 40007. címen folytatódik, azaz a rutinnak vége. BASIC-ben ezt a következőképpen szemléltethetjük volna:

```
10 LET C=15
20 LET C=C-1
30 LET A=0
40 IF C<>0 THEN GO TO 20
50 STOP
```

6. Az előjel jelzőbit állapotától függő ugrás

Itt csak két utasítást találunk. Ezek abban az esetben hajtanak végre ugrást, ha az előjel jelzőbit az utasításban megadott állapotú. Az előjel jelzőbit (SIGN FLAG - POSITIVE = P, MINUS = M) az 'F' regiszter 7. bitje. Általában az eredmény legnagyobb helyiértékű bitjének állapota szerint áll be. Értéke pozitív számoknál zérus, míg negatív számoknál 1. Az előjel jelzőbit tehát pont fordított eredményt tanúsít. Állapotát az előzőekben felsorolt utasítások a már említett módon befolyásolják.

F2	X	Y	242,x,y	JP P,CIM
FA	X	Y	250,x,y	JP M,CIM

ahol x,y (Xh, Yh) a 'CIM' alsó/felső byte-ja.

Az első esetben akkor történik ugrás, ha az eredmény pozitív, a második esetben pedig akkor, ha negatív.

7. A túlszordulás/paritás jelzőbit állapotától függő ugrás

Itt is két utasítást különböztetünk meg. Ezek akkor hajtanak végre ugrást, ha a túlszordulás/paritás jelzőbit az utasításban megadott állapotú. A túlszordulás/paritás jelzőbit (OVERFLOW/PARITY FLAG = P/V) az 'F' regiszter 2. bitje. Szerepe kettős. Több ízben az eredmény a túlszordulásnak megfelelő, míg más utasítások végrehajtása esetén a jelzőbit a paritás-ellenőrzés eredményének megfelelően befolyásolódik.

E2	X	Y	226,x,y	JP PO,CIM
EA	X	Y	234,x,y	JP PE,CIM

ahol x,y (Xh, Yh) a 'CIM' alsó/felső byte-ja.

Ezek az utasítások ritkán használatosak gépi kódú programokban, elsősorban azért, mert más utasításokkal is kiválthatók.

A következőkben a ciklus-szervezéssel, a stack műveletekkel, a bit-léptető utasításokkal fogunk foglalkozni. Lassan belekezdhetünk egyszerű gépi kódú programjaink megtervezésébe is. Akik úgy érzik, már kaptak elég ihletet az alapok elsajátításához, azoknak javasoljuk, tanulmányozzák a 'Programozástechnika' témakörökben szereplő listákat, sok hasznos fogást elleshetnek a rutinok szerkezetéből, nem beszélve arról, hogy az eddig megismert utasítások működésére is találhatnak kiváló mintapéldákat.